11-18-2005

Hyperspectral sub-pixel target detection using hybrid algorithms and Physics Based Modeling

Emmett J. Ientilucci

Hyperspectral Sub-Pixel Target Detection Using Hybrid Algorithms and Physics Based Modeling

by

Emmett J. Ientilucci

A.A.S.Monroe Community College, 1986
B.S. Rochester Institute of Technology, 1996
M.S. Rochester Institute of Technology, 1999

A dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the Chester F. Carlson Center for Imaging Science
Rochester Institute of Technology

November 18, 2005

Signature of the Author Emmett Ientilucci

Accepted by Name Illegible 11/18/05
Coordinator, Ph.D. Degree Program Date

CHESTER F. CARLSON CENTER FOR IMAGING SCIENCE
ROCHESTER INSTITUTE OF TECHNOLOGY
ROCHESTER, NEW YORK

CERTIFICATE OF APPROVAL

Ph.D. DEGREE DISSERTATION

The Ph.D. Degree Dissertation of Emmett J. Ientilucci
has been examined and approved by the
dissertation committee as satisfactory for the
dissertation required for the
Ph.D. degree in Imaging Science

John R. Schott
Dr. John R. Schott, Dissertation Advisor

Harvey E. Rhody
Dr. Harvey E. Rhody

John P. Kerekes
Dr. John P. Kerekes

P. Bajorski 9/24/05
Dr. Peter Bajorski

11/11/2005
Date

DISSERTATION RELEASE PERMISSION
ROCHESTER INSTITUTE OF TECHNOLOGY
CHESTER F. CARLSON CENTER FOR IMAGING SCIENCE

Title of Dissertation:

Hyperspectral Sub-Pixel Target Detection Using Hybrid Algorithms and Physics Based Modeling

I, Emmett J. Ientilucci, hereby grant permission to Wallace Memorial Library of R.I.T. to reproduce my thesis in whole or in part. Any reproduction will not be for commercial use or profit.

Signature Emmett Ientilucci 11-18-05
Date

Hyperspectral Sub-Pixel Target Detection Using Hybrid Algorithms and Physics Based Modeling

by

Emmett J. Ientilucci

Submitted to the
Chester F. Carlson Center for Imaging Science
in partial fulfillment of the requirements
for the Doctor of Philosophy Degree
at the Rochester Institute of Technology

Abstract

This thesis develops a new hybrid target detection algorithm called the Physics Based-Structured InFeasibility Target-detector (PB-SIFT) which incorporates Physics Based Modeling (PBM) along with a new Structured Infeasibility Projector (SIP) metric. Traditional matched filters are susceptible to *leakage* or false alarms due to bright or saturated pixels that appear target-like to hyperspectral detection algorithms but are not truly target. This detector mitigates against such false alarms.

More often than not, detection algorithms are applied to atmospherically compensated hyperspectral imagery. Rather than compensate the imagery, we take the opposite approach by using a physics based model to generate permutations of what the target might look like as seen by the sensor in radiance space. The development and status of such a method is presented as applied to the generation of target spaces. The generated target spaces are designed to fully encompass image target pixels while using a limited number of input model parameters. Evaluation of such target spaces shows that they can reproduce a HYDICE image target pixel spectrum to less than 1% RMS error (equivalent reflectance) in the visible and less

than 6% in the near IR. Background spaces are modeled using a linear subspace (structured) approach characterized by basis vectors found by using the maximum distance method (MaxD).

The SIP is developed along with a Physics Based Orthogonal Projection Operator (PBosp) which produces a 2 dimensional decision space. Results from the HYDICE FR I data set show that the physics based approach, along with the PB-SIFT algorithm, can out perform the Spectral Angle Mapper (SAM) and Spectral Matched Filter (SMF) on both exposed and fully concealed man-made targets found in hyperspectral imagery. Furthermore, the PB-SIFT algorithm performs as good (if not better) than the Mixture Tuned Matched Filter (MTMF).

Acknowledgements

We all have milestones in our lives and pursuing my Ph.D. was one of them. I can remember the day that changed all subsequent days like it was yesterday. After completion of my M.S. degree, I was to ask Dr. John Schott about potential employment opportunities. Since I needed to stay local at the time and no one was hiring, he suggested we talk about other endeavors in his car on our way to Toys "R" Us, Inc. That's when he suggested entering the doctoral program, which I (previously) never considered. I took the weekend to think about it and some number of years later, here I am writing this commentary in my completed dissertation. That 10 minute conversation changed the rest of my life. Thanks Dr. Schott for your years of input and mentoring on both personal and academic issues.

I would also like to acknowledge many of my colleges and friends who sat with me over the years to discuss such matters related to this body of work. Specifically, these these include Alex Tyrrell, Dave Messinger, Rolando Raqueno, and Scott Brown. Thanks for lending me your ears.

Thanks goes to Cindy Schultz who pretty much runs our department and provided general support. Thanks for helping me with all the logistical organization of committee meetings, etc. Getting them together on the same date was like herding a bunch of stray cats! A special thanks to my girlfriend Sherry for supporting me throughout this dissertation.

Finally, I would like to thank my committee members; Dr. Peter Bajorski, Dr. John Kerekes and Dr. Harvey Rhody who provided additional insight and took the time to read my dissertation. You all had something valuable to offer. Thanks.

This work is dedicated to Sherry, my parents; Emmett and Jean, the memory of my late sister, Lynn and my nephew, Justin.

Nomenclature

Symbol	Description
a	Abundance or fractional weight scalar
$\mathbf{a}$	Abundance or fractional weight vector
$\mathbf{d}$	Un-target-like vector
$E[\mathbf{x}]$	Expected value of $\mathbf{x}$
$E_s^{'}(\lambda)$	Exoatmospheric irradiance
FA	False alarm
GLR	Generalized likelihood ratio
$\mathcal{H}_0$	Null hypothesis
$\mathcal{H}_1$	Alternative hypothesis
INF	Infeasibility
$L_d(\lambda)$	Spectral downwelled radiance
$L_s(\lambda)$	Spectral radiance
$L_u(\lambda)$	Spectral upwelled radiance
$\mathbf{L}$	Radiance vector
$\mathcal{L}(\mathbf{x})$	Likelihood ratio
$\mathbf{\Lambda}$	Diagonal matrix
LR	Likelihood ratio
M	Miss

MLE	Maximum likelihood estimate
MTMF	Mixture tuned matched filter
$\boldsymbol{\mu}_b$	Background mean vector
$\boldsymbol{\mu}_t$	Target mean vector
$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	Normal distribution with mean vector $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
$\mathbf{P}$	Projection operator
$\mathbf{P}^{\perp}$	Orthogonal projection operator
$r(\lambda)$	Spectral reflectance
ROC	Receiver operating characteristic (ROC) curve
$\boldsymbol{\Sigma}$	Covariance matrix (population)
$\mathbf{S}$	Covariance matrix
$\sigma^{'}$	Solar zenith angle
$\mathbf{t}$	Target spectrum (bold for vector)
T_{ACE}	Adaptive coherence estimator
T_{AMF}	Adaptive matched filter
T_{AD}	Anomaly detector
T_{PBosp}	Physics based orthogonal subspace projector
T_{CEM}	Constrained energy minimization
T_{FTMF}	Finite tuned matched filter
T_{GLR}	Generalized likelihood ratio detector
T_{Gosp}	Generalized orthogonal subspace projector
T_{Gosp_s}	Generalized orthogonal subspace projector, for single vector
T_{MF}	Matched filter detector or Fisher's linear discriminant
T_{MF_mc}	Matched filter detector on mean-centered data
T_{MF_norm}	Normalized matched filter detector
T_{MFWN}	Matched filter for white noise

T_{MSD}	Matched subspace detector
T_{OSP}	Orthogonal subspace projector
T_{QD}	Quadratic detector
T_{SIP}	Structured infeasibility projector
T_{SMF}	Spectral matched filter
$\mathbf{x}$	Pixel spectrum (bold for vector)
$\tau_1(\lambda)$	Spectral atmospheric transmission sun-to-target path
$\tau_2(\lambda)$	Spectral atmospheric transmission target-to-sensor path
$\mathbf{w}$	Image vector in normalized, mean-centered, whitened space
$\mathbf{w}$	Vector of residuals to account for noise and lack of fit
$\mathbf{w}_t$	Target vector in normalized, mean-centered, whitened space
WV	Water vapor

Contents

List of Figures

List of Tables

"Anybody who has been seriously engaged is scientific work of any kind realizes that over the entrance to the gates of the temple of science are written the words: 'Ye must have faith."

Max Planck

Introduction

1.1 Remote Sensing and Spectroscopy

Remote sensing, for all practical purposes, is the field of study associated with extracting information about an object without coming into physical contact with it. A definition such as this can include a multitude of disciplines such as medical imaging, astronomy, vision, sonar, and earth observation from above via an aircraft or space satellite. For the most part, remote sensing is often thought of in the latter context, viewing and analyzing earth from above.

The field of remote sensing is has been evolving for over 50 years. This goes back to the late 40s and early 50s when some of the first satellite photographs were being recovered from V-2 launches. During this time the satellite imaging of the earth

developed hand in hand with the international space program. Following the first manmade satellite launch of Sputnik 1 on October 4, 1957, photographs and video images were acquired by U.S. Explorer and Mercury programs and the Soviet Unions LUNA series. Just 3 years after Sputnik 1's debut (April 1960), the U.S. initiated its spaced-based reconnaissance program acquiring high-resolution photographic images from space [38].

Today remote sensing is used in a variety of applications such as environmental mapping, global change research, geological research, wetlands mapping, assessment of traceability, plant and mineral identification and abundance estimation, target detection and classification, and crop analysis. What makes the study of these application areas possible is the fact that materials that make up objects in a scene (*i.e.*, vehicles, wheat fields, body of water, etc.) can be characterized due to the fact that they reflect, emit, scatter, and absorb electromagnetic (EM) radiation differently. One can take advantage of this fact and say that, ideally, each material therefore has a unique characteristic feature. If we are looking at this interaction in many regions of the EM spectrum then we go on to further say that each material has a unique *spectral signature*. The analysis and interpretation of such spectral signatures is called *spectroscopy*.

This material signature is usually propagated through the atmosphere where it falls incident on an imaging systems *sensor*. Today, most imaging system sensors are of a solid state design. This enables one to design a system (in conjunction with a dispersing element) that can simultaneously capture spatial as well as spectroscopic information about a piece of real estate below. These systems are called *imaging spectrometers* and can be essentially divided into those that use one to ten or so spectral channels (multispectral) and those with tens to hundreds of spectral

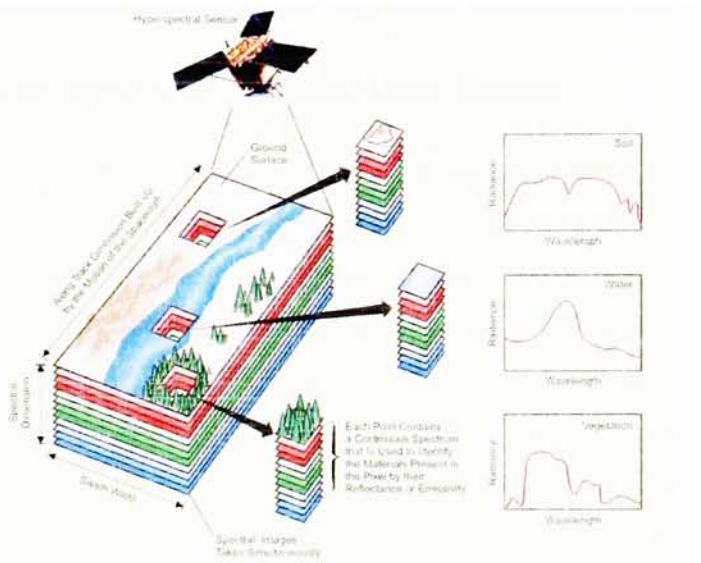


Figure 1.1: Overview of collection capabilities of a hyperspectral sensor system. Shown are materials with distinctive spectroscopic characteristics obtained through the many band or channels the instrument collects [49].

channels (hyperspectral). An example of such a system is shown in Figure 1.1.

The field of imaging spectrometery has been under development since the 1970s as a means of identifying and mapping earth resources. As a result, hyperspectral data, in particular, permit the expansion of detection and classification activities to targets previously unresolved in multispectral images. Examples of such hyperspectral imaging systems are the Airborne Visible InfraRed Imaging Spectrometer (AVIRIS) [16] which images in 224 bands and the HYperspectral Digital Imagery Collection Experiment (HYDICE) [17] which images in 210 bands. Both systems image the EM spectrum from 400 to 2500nm.

1.2 Hyperspectral Application Areas

Of the many applications areas in remote sensing, hyperspectral remote sensing tends to focus on topics in three major categories, (*i*) anomaly detection, (*ii*) target recognition and detection, and (*iii*) background characterization. *Anomaly detection* seeks to find signatures of unusual materials, as compared to background materials, without *a priori* knowledge of the target. In contrast, *target detection* algorithms use known information about a material of interest and seek to find such a material(s) in the scene. The *a priori* information may originate from a spectral library or be image derived. In *target recognition*, we not only detect the target(s) but we assign it a label or name. This method of detection is called *classification*. Classification is usually a name given when one is dealing with grouping a large number of pixels into multiple classes. For target detection, the number of objects one seeks to find is usually very sparse and man-made in nature. The last category, *background characterization*, deals with the characterization of the overall scene background. Examples of this include, terrain categorization, water quality assessment, atmospheric compensation, and plume detection and quantification. In this research, we focus on the development of target detection algorithms that use hyperspectral data captured with imaging spectrometers.

1.3 Hyperspectral Target Detection: Considerations

There are many issues that can complicate the success or outcome of the above mentioned algorithms. One such issue is *atmospheric compensation*. In most cases, one atmospherically compensates imagery to ground leaving reflectance before applying a target detection algorithm. This is because *a priori* information about targets of

interest is usually measured in reflectance units. In this case the success of the detection algorithm is highly dependent on the validity of the compensation algorithm. The method and type of compensation is a significant part of hyperspectral remote sensing and is the subject of ongoing research. In this research we will be taking the approach of developing and applying algorithms on uncompensated imagery. Rather than "back out" the atmosphere to gain access to the ground reflectance target spectrum we simply propagate the target through the atmosphere using a physics based model and generate a probabilistic target space that resembles what the target might look like as seen by the imaging sensor. That is, a target sensor reaching *radiance space* is developed that can be used in a detection scheme.

Another issue that needs to be considered is variability in the data. In general, materials in hyperspectral data do not exhibit deterministic behavior. Rather, a single material will exhibit a fair amount of random variability that needs to be accounted for in the detection model. The variability can arise from poor atmospheric compensation, spectrometer instrument error, sensor noise, self shadowing, adjacency effects and geographic location. Additionally, variability can arise from natural variation in the target or variability due to lack of specific knowledge about the target. The most popular techniques to characterize this variability involve either vector algebra (geometry) or statistical descriptors. In the geometric sense we use *endmembers*, whereas means and covariances are used as statistical descriptors. Examples include probability density (statistical) and (linear) mixing models (geometric).

Lastly, there is the issue of full versus mixed pixels. Depending on the sensor ground resolved distance (GRD), pixels will either be 100% filled or partially filled with a material of interest. Therefore, one needs to account for this effect using an

appropriate model.

1.4 Research Overview: Work Statement

Most detection models are based on the traditional matched filter and describe the background variability through use of geometric or stochastic means. This research explores the development of a geometric algorithm that factors in a physically based target model to account for target variability while using an additional metric that aids in rejecting false alarms. This geometric approach is very similar to the stochastic Mixture Tuned Matched Filter (MTMF) which uses a spectral matched filter and an infeasibility metric to reduce false alarms. An overview of the research into physics based modeling and algorithm development is provided below.

- Full literature search and overview of relevant concepts and methods as they relate to physics based modeling and target detection.
- Improved physics based modeling of a target space. This includes investigation on what parameters to use when generating a target space, how they are used, and what values are appropriate when using them.
- Incorporate improvements to current usage of physics based modeling. This includes any modifications to the sensor-reaching radiance equation.
- Develop a new (hybrid) detector that can utilize results from physics based modeling.
- Develop a geometric equivalent to the infeasibility concept.
- Derive the Mixture Tuned Matched Filter (MTMF) algorithm used in ENVI.

- Develop a new geometric detector that is based on the MTMF.
- Explore the behavior of 2D detector spaces.
- Compare results of new physics based detectors to the Spectral Angle Mapper (SAM) and the Spectral Matched Filter (SMF) in conjunction with atmospheric compensation algorithms such as ELM, ATREM and FLAASH.

"Facts are the air of scientists. Without them you can never fly."

Linus Pauling

"The fewer the facts, the stronger the opinion."

Arnold H. Glasow

2

Background

In this background section we describe many of the approaches used in traditional target detection. We start off with an explanation of signal detection theory and how it relates to target detection through development of the hypothesis test. This leads us to the derivation of the likelihood and generalized likelihood ratio tests. We then explore methods to characterize background spaces in a stochastic sense followed by appropriate statistical models. This is a segue into geometrical (end-members) methods to characterize the background followed by a description of the linear mixing model. Subsequent sections describe approaches to target detection (both in a geometric and statistical sense) for fully resolved and mixed pixels.

We then introduce the concept of target spaces and physics based modeling through description of an illumination invariant technique called the invariant al-

gorithm. Finally, we introduce hybrid algorithms that take advantage of statistical and geometric descriptors.

2.1 Signal Detection Theory

Detection theory is a generalized term used to categorize decision making about whether or not a particular event occurs. For example, in radar one might be interested in determining the presence or absence of an approaching aircraft, which may be obscured by background noise. Here we choose between two possible cases in seeking out the aircraft: (*i*) the aircraft and noise are present or (*ii*) noise only is present. This decision making process can be thought of as a *binary hypothesis testing* problem. If the nature of the data is random, with noise as an example, then a statistical approach is necessary. Here, the binary hypothesis testing problem turns into a *statistical* hypothesis testing problem. The typical target detection chain that creates such binary detection maps can be seen in Figure 2.1.

In general, the simplest detection problem is one that involves a known deterministic signal and contains background noise that is Gaussian with a known probability density function (PDF). The problem becomes more difficult as the deterministic signal becomes random with an unknown distribution and the Gaussian assumptions become non Gaussian with unknown PDF's. These latter conditions all decrease detection performance due to the fact that less is known about the characteristics of the signal and noise.

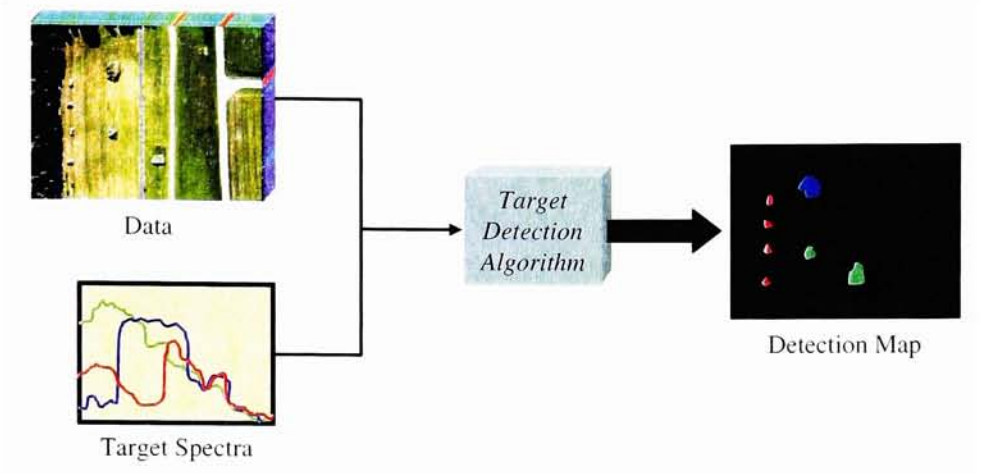


Figure 2.1: Typical target detection process showing a detection map for the 3 input target spectra.

2.1.1 Hypothesis Testing and the Likelihood Ratio (LR)

Lets say we are given an observed pixel spectrum $\mathbf{x}$ such that

$$\mathbf{x}^T = [x(\lambda_1), x(\lambda_2), \cdots, x(\lambda_p)] \tag{2.1}$$

for p-spectral bands and we wish to decide if the pixel has an interesting "target" in it. Given this situation, we establish the following binary hypothesis test for the pixel:

$$\mathcal{H}_0: \text{ target absent}$$
$$\mathcal{H}_1: \text{ target present}$$

where $\mathcal{H}_0$ is referred to as the *null hypothesis* and $\mathcal{H}_1$ as the *alternative hypothesis*. For illustrative purposes, we present a univariate example of this binary hypothesis test. If we assume x is a realization of a multivariate Gaussian random variable, then we will assume the PDF of x from a single band comes from either $\mathcal{N}(\mu_0, \sigma^2)$ or $\mathcal{N}(\mu_1, \sigma^2)$, as shown in Figure 2.2.

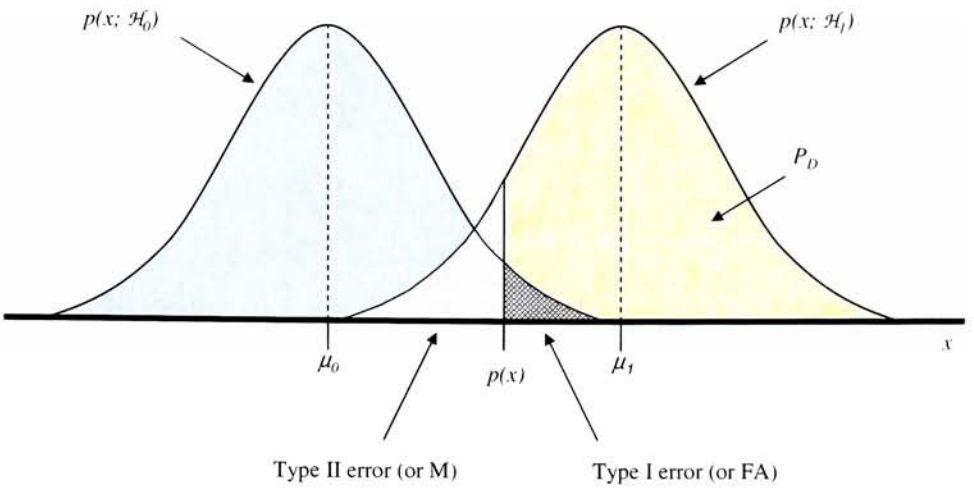


Figure 2.2: Probability density functions for hypothesis testing problem showing errors and probability of detection P_D.

In the event that we decide $\mathcal{H}_1$ (target present) but $\mathcal{H}_0$ is true (target is actually absent), we make what is called a *Type I error*. This is typically termed a *false alarm* (FA) by engineers. On the other hand, if we decide $\mathcal{H}_0$ but $\mathcal{H}_1$ is true, we make a *Type II error*. The engineering community refers to this type of error as a *miss* (M) (*cf.* Figure 2.2). In general, it is well known that detectors based on the likelihood *ratio* (LR) test have certain advantages over analysis of the probability functions individually [33]. The LR minimizes the risk associated with incorrect decisions as well as leads to detectors that are optimum for a wide range of performance criteria. This ratio is usually defined as

$$\mathcal{L}(\mathbf{x}) = \frac{p(\mathbf{x}; \mathcal{H}_1)}{p(\mathbf{x}; \mathcal{H}_0)}. \tag{2.2}$$

If $\mathcal{L}(\mathbf{x})$ exceeds a threshold γ, then the alternative hypothesis (target present) is selected as true. Typically, the quantity $\mathcal{L}(\mathbf{x})$ with threshold γ is referred to as a

detector or *detection statistic*, the former of which is a label used by the engineering community [23].

If the probability density functions in Eq. (2.2) are completely known, then the test is called a *simple* hypothesis test. In the event that the PDF's are not completely known, then we have what is called a *composite* hypothesis test.

2.1.2 The Neyman-Pearson Approach

It is always of great desire to select the threshold γ, such that one maximizes the number of correct detections while minimizing the number of target misses and false alarms. In the event that we have a *simple* hypothesis test we can choose a threshold based on either the Neyman-Pearson theorem or the Bayesian approach, which is based on minimization of the Bayes risk most often seen in classification problems. In the later, we choose a threshold that leads to minimize both false alarms (FA) and misses (M). However, in detection applications, where the probability of occurrence of a target is very small, minimization of the error probability (FA and M) is not a good criterion of performance, because the probability of error can be minimized by classifying every pixel as background [33]. A better approach is to maximize the probability of choosing $\mathcal{H}_1$ when indeed it is true (detection), while not exceeding a "fixed" probability of choosing $\mathcal{H}_1$ when it is false (false alarm). This approach is called the Neyman-Pearson theorem [23] which simply maximizes the probability of detection (P_D) given a fixed value α for the probability of false alarms (P_{FA}). Therefore, we decide $\mathcal{H}_1$, for a given $P_{FA} = \alpha$, if

$$\mathcal{L}(\mathbf{x}) = \frac{p(\mathbf{x}; \mathcal{H}_1)}{p(\mathbf{x}; \mathcal{H}_0)} > \gamma. \tag{2.3}$$

When working with remote sensing data typical per-pixel values for α range from 10^{-4} to 10^{-6}.

2.1.3 Generalized Likelihood Ratio

In practice, information about the PDF's under the two hypotheses ($\mathcal{H}_0$ and $\mathcal{H}_1$) may not be known. In this case we cannot use the Neyman-Pearson method described above. Instead, we replace the unknown parameters in Eq. (2.3) with their maximum likelihood estimates (MLE). This approach is called the generalized likelihood ratio (GLR) test and produces a family of, what are typically called, *adaptive* detectors. These detectors are termed adaptive because the unknown parameters must be estimated from the actual image data.

2.2 Estimation of Target and Background Spaces

In the previous section we introduced the concept of a likelihood ratio based on the Neaman-Pearson criteria. If the distributions in the two hypotheses contain unknown parameters, then the detector was termed "adaptive" and called a generalized likelihood ratio test. In this case, the unknown parameters must be estimated from the background or target space. There are many techniques in estimating these parameters all of which depend on the type of detection model used. If the model requires means and covariance's then a statistical approach is necessary. However, if we use a subspace or mixing model, then the matrices in the model are usually populated with endmembers found using geometric techniques.

2.2.1 Stochastic Methods

More often than not, we are trying to characterize background (or target) variability due to effects such as atmospheric conditions, noise, or variability in the material itself. In the event that the background or target does not vary (*i.e.,* is deterministic) then there would be no need for characterization of targets or background via stochastic methods. This is usually not the case. There are many different ways to characterize these target and background spaces, some of which are discussed below.

Local (Windowing) and Global Approaches

A straight forward approach to characterizing a target or background is to simply compute the first and second moments based on the *entire* space or data set. That is, the overall mean $\boldsymbol{\mu} = E[\mathbf{x}]$, and covariance $\boldsymbol{\Sigma}$ given by

$$\boldsymbol{\Sigma} = E[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T]. \tag{2.4}$$

It is clear that this descriptor becomes less accurate as the overall target or background "class" becomes less homogenous or multi-modal. Unfortunately, most real-world image backgrounds are highly non-homogenous or "cluttered". Though this is the case, many people still use this approach as a first order approximation or guess as to how the background or target is behaving.

Rather than generate statistics based on the entire image, a reasonable next step might be to sub-section or spatially *window* parts of the image. This assumes there is some spatial structure in what one is trying to estimate as opposed to data that may only contain spectra without spatial structure (*e.g.,* a spectral database or library for a single material). Windowing would be more commonly employed when estimating a background mean and covariance. The essences of this approach is to

simply generate a window of a given size, much smaller than the overall image size, and slide it across the image in a predictable manor. The idea is to reduce the overall variability by confining our estimate to a local region that is more homogenous than the entire image.

Classification and Clustering

Thus far we have talked about gathering information about variability using a global or local spatial technique. We could also classify or cluster the data based on similarity of the spectra in the image then compute the appropriate moments from this result. Classification is more concerned with *separating* distinct sets of objects (or observations) and with *allocating* new objects (observations) to previously defined groups. Clustering is distinct from classification in that it is a more primitive technique in that no assumptions are made concerning the number of groups or the group structure.

Classification or discrimination algorithms include measures such as minimum distance to a class mean (MDM) [46]. Here we simply compute the length between an unknown pixel under test to the mean of each class. The unknown pixel goes into the class with the shortest length. Additional examples of classification algorithms include Gaussian maximum likelihood (GML) [46] and Fisher's linear discriminant [20].

Clustering algorithms try to find natural groupings of items or variables. They tend to fall into two categories called hierarchical or non-hierarchial. Within the hierarchical category we have what are called, for example, linkage methods (single, complete, and average linkage). Linkage methods are suitable for clustering items, as well as variables. Non-hierarchical clustering techniques are designed to group items,

rather than variables, into a collection of k-clusters. The number of clusters may either be specified in advanced or determined as part of the clustering procedure. An example of a non-hierarchical clustering algorithm would be the well known k-means [8] algorithm.

Stochastic Expectation Maximization (SEM)

Clustering is also performed through use of the stochastic expectation maximization (SEM) algorithm [37]. This is a spectral clustering technique for classifying spectral data that involves iterative estimation of a Gaussian mixture fit to spectral data. The SEM uses a statistical model for the scene, where each hyperspectral pixel may come from a different Gaussian distribution (class) where there is a finite number of classes. The method fits a Gaussian mixture model (GMM) to the background clutter data. Pixels in a region are assigned to one of the classes based on estimates of *a posteriori* probability of the class, given the sample. The assignment to the classes is done according to the *a posteriori* probabilities of the classes given the pixel value, thus the "stochastic" in the algorithm name. The statistics of the class are then computed and the assignments are repeated. This procedure is applied iteratively until the data are sufficiently characterized by the model. Segmentation maps can be produced by assigning individual pixels to a class using a maximin *a posteriori* (MAP) classifier.

2.2.2 Stochastic Models

Probabilistic Model or Gaussian Mixture Model (GMM)

We can represent the clustered or classified information as a probabilistic density model. Here, we model each datum as an observation of a random vector having

one of several (M) possible multivariate Gaussian distributions. That is

$$f(\mathbf{x}) = \sum_{i=1}^{M} \pi_i \, f(\mathbf{x}; \mu_i, \Sigma_i) \tag{2.5}$$

where each observed spectrum is generated by first selecting a spectrally homogeneous class i with probability π_i, where $\pi_i \geq 0$ and

$$\sum_{i=1}^{M} \pi_i = 1 \tag{2.6}$$

and then generating a spectrum $\mathbf{x}$ according to the conditional probability law $f(\mathbf{x}; \mu_i, \Sigma_i)$ specified by the parameter vectors μ_i, Σ_i . It is noted that the component densities $f(\mathbf{x}; \mu_i, \Sigma_i)$ are usual multivariate normal distributions defined by

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{2.7}$$

where p is the number of spectral bands and $|\boldsymbol{\Sigma}|$ is the determinant of $\boldsymbol{\Sigma}$. The parameter vectors could also be estimated using the stochastic expectation maximization (SEM) technique outlined in Sec. 2.2.1.

Stochastic Mixing Model (SMM)

If we wish to look at the intermediate *mixing* of these clusters in a stochastic sense, then we can use the stochastic mixing model (SMM) [51]. This approach suggests that the background clutter is comprised of clusters called *hard* endmembers. Fractional mixtures of hard endmembers can form a dispersed continuum of spectral vectors between the hard endmembers (*cf.* Figure 2.3). These clusters (both hard and fractional) are described using a unique multivariate Gaussian probability density. Image pixels belonging to a given class are assumed to be independently drawn from the density that defines that class.

To specify a SMM one needs to define the number of hard classes C_h and a desired fractional resolution Δf of the mixed classes to be generated. The hard classes are assumed to be normal and are described as

$$\mathcal{N}(\mathbf{m}_c, \mathbf{S}_c) \text{ for } c = 1, 2, \ldots, C_h \tag{2.8}$$

where $\mathbf{m}$ is the mean vector with associated covariance $\mathbf{S}$.

A total of C_m additional mixed classes can then be defined by a set of fractional vectors

$$\mathbf{f}_c = [f_c(1), f_c(2), \ldots, f_c(C_h)]^T \tag{2.9}$$

that satisfy the constraints $\sum_{i=1}^{C_h} f_c(i) = 1$ with $0 < f_c(i) < 1$.

It can be shown that a SMM defined with C_h hard classes and a mixture fraction resolution of Δf generates $C = L(L+1)\cdots(L+C_h-2)/(C_h-1)!$ feasible classes, where $L = 1+1/\Delta f$. Of these C classes, $C_m = C - C_h$ represent mixed pixel classes. These mixed pixel classes have mean and covariance parameters given by

$$\mathbf{m}_c = \sum_{i=1}^{C_h} f_c(i)\, \mathbf{m}_i \tag{2.10}$$

$$\mathbf{S}_c = \sum_{i=1}^{C_h} f_c^2(i)\, \mathbf{S}_i \tag{2.11}$$

To illustrate this approach consider the two class ($C_h = 2$) problem shown in Figure 2.3. If we specify a mixing resolution of $\Delta f = 1/6$, then we will have a total of 7 clusters. Two of them representing hard classes and five representing mixed pixel classes. The associated fractional vectors for the 2-band, 2-class example would then be

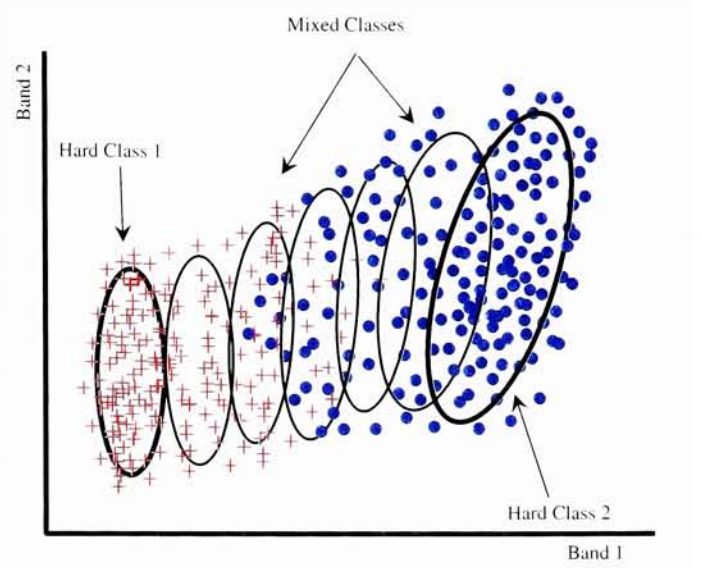


Figure 2.3: Illustration of the stochastic mixing model (SMM) concept for a simple 2 class, 2 band case. Drawn are isoprobability contours illustrating clusters associated with hard-endmember classes and mixed pixel classes.

$$\mathbf{f}_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \mathbf{f}_2 = \begin{bmatrix} 1/6 \\ 5/6 \end{bmatrix}, \mathbf{f}_3 = \begin{bmatrix} 2/6 \\ 4/6 \end{bmatrix}, \ldots, \mathbf{f}_7 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{2.12}$$

Note that if the number of hard classes is larger and we allow mixtures of more than 2 hard classes, the number of mixed classes grows rapidly. Thus the authors suggest that this approach is most appropriate for low contrast targets (*i.e.*, where simpler approaches fail) and where the clutter can be reduced to a small number of hard endmember classes.

2.2.3 Geometric Methods

The previous sections are all based on describing the background (or target) spaces through use of descriptive statistics. An alternative, and equally valid, approach is

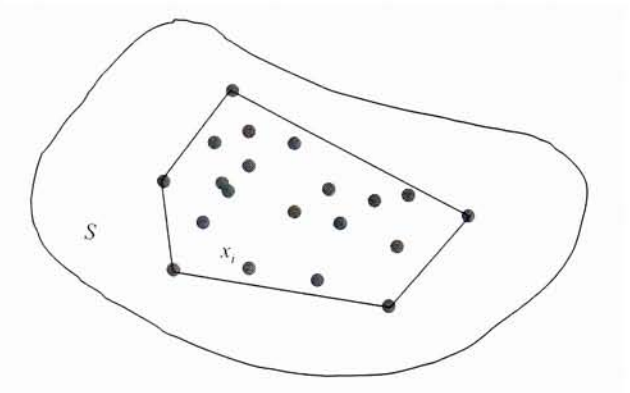


Figure 2.4: Illustration of the convex hull as being the minimal convex set encompassing the data.

to describe the data with vector geometry which uses the concept of "endmembers" which can be thought of as points in the data that represent "pure" materials of interest. These endmembers attempt to encapsulate the data in what is known as a convex hull.

In mathematics, the convex hull for an object or set of objects is the minimal convex set containing the given objects. It is the *minimal* convex set because the convex hull is a subset of any convex set which contains the given objects. If S is some set, the convex hull of S can be described as the set of points of the form

$$\sum_{i=1}^{n} \varpi_i x_i \tag{2.13}$$

where n is an arbitrary natural number, the numbers ϖ_i are non-negative and sum to 1, and the points x_i are in S. This concept is illustrated in Figure 2.4 for points lying in a plane.

Many of the available algorithm for computing the convex hull only apply to 2 dimensional spaces [11]. Still others have a natural extension to 3 dimensions, such as the divide-and-conquer algorithm [39]. Since hyperspectral data can contain

hundreds of dimensions, the application of such algorithms is inappropriate. In the following section we illustrate some of the more popular methods for estimation of discrete points (or endmembers) that describe the convex hull.

Pixel Purity Index (PPI)

A simple, but effective, approach is called the pixel purity index (PPI) [6]. Here pixel vectors are repeatedly projected into random directions. The coordinates of the direction vectors are generated using a standard normal distribution. Once all projections on a given random vector are calculated, two extreme projections are identified (the shortest and the longest one). Each of the two pixels associated with the two extreme projections receives one extremity score. This process is repeated many times and a desired number of pixels with the top extremity scores are identified as the set of endmembers.

One problem with the PPI method is that clusters of pixels that are close to each other tend to receive large extremity scores. Consequently, identification of those similar pixels, through cluster analysis, is needed so that only one pixel per cluster is represented in the final set of endmembers.

Singular Value Decomposition (SVD)

Another approach that generates basis vectors is called the singular value decomposition (SVD). Unlike PPI or MaxD (described below) which produce *native* endmembers, the SVD produces orthonormal eigenvectors which form a minimum spanning set of basis vectors that point in the directions of maximum variability. In the SVD formulation we define a $p \times N$ matrix $\mathbf{Y}$ of all image pixels (given as columns) and consider the singular value decomposition of $\mathbf{Y}$:

$$\mathbf{Y} = \mathbf{UDV}^T \tag{2.14}$$

where $\mathbf{U}_{p\times p} = [u_j]_{j=1,\ldots,p}$ is the matrix of eigenvectors of $\mathbf{YY}^T$ and $\mathbf{D}$ is the diagonal matrix of singular values σ_i, such that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p \geq 0$, noting that the singular values are the square roots of the eigenvalues (λ_i) from $\mathbf{YY}^T$ and $\mathbf{Y}^T\mathbf{T}$ (*i.e.*, $\sigma_i = \sqrt{\lambda_i}$). One can then use the first r columns of $\mathbf{U}$, that is, $\mathbf{B} = [u_1, \ldots, u_r]$, as the basis vectors. Criteria for choosing the number of basis vectors r are mostly based on the percent of variability explained by the first r vectors. Since the SVD is very efficient in capturing the directions (vectors) explaining most of the variability, a relatively small number of basis vectors tends to explain more than 99.99% of the overall variability. Unfortunately, it is difficult to decide how much explained variability is sufficient for the purpose of target detection. A more detailed explanation of the SVD can be found in Appendix A.

Maximum Distance Method (MaxD)

This section introduces another geometric method that can be used for the selection of endmembers, given a hyperspectral image. The techniques relies on the fact that a hyperspectral data set can be well approximated by a convex set made up of linear combinations of endmembers of a given scene where the weights are all positive and sum to one. Ideally, the endmembers selected should form the best fitting simplex around the data in an n-dimensional space. The theoretical basis of this method is based on the fact that for any given point in a simplex, a point with maximum distance to the given point must be one of the vertices of the simplex, hence the name, maximum distance method (MaxD).

This is a relatively new approach to endmember selection originally suggested

by Lee [31] and further investigated by Schott [48]. The method consists of finding *native* (*i.e.*, vectors that are in the original image space) endmember vectors that best approximate a simplex defining the target/background subspace. The technique starts with identifying two pixels, one with the largest magnitude vector (denoted by $\mathbf{v}_1$) and one with the smallest magnitude (denoted by $\mathbf{v}_2$). Next, all pixel vectors are projected along $\mathbf{v}_1 - \mathbf{v}_2$ onto the subspace orthogonal to $\mathbf{v}_1 - \mathbf{v}_2$ (*cf.* Figure 2.5). In this projection, both $\mathbf{v}_1$ and $\mathbf{v}_2$ project on the same point (which we will call $\mathbf{v}_{12}$). Then, the distance between $\mathbf{v}_{12}$ and the remaining projections are calculated. The pixel with the maximum distance to $\mathbf{v}_{12}$ is the third endmember (denoted by $\mathbf{v}_3$). All projected points are now projected along $\mathbf{v}_{12} - \mathbf{v}_3$. The resulting endmember is denoted by $\mathbf{v}_{123}$. The process is repeated until a desired number of endmembers is identified. If this process is continued until $(p + 1)$ endmembers are identified, all projected points reduce to one point, and the process can no longer be continued. That is, we can identify up to $(p + 1)$ endmembers using MaxD, which is not a limitation in practice when working with hyperspectral images. Additionally, the MaxD method is very fast computationally and is fully automated.

2.2.4 Geometric Models

Linear Mixing Model (LMM)

One of the more popular geometric models, for sub-pixel target detection, is the linear mixing model (LMM). This model assumes that the pixel of interest is comprised of a linear combination of unique deterministic spectra which we call endmembers. The general equation for assuming a linear mixture of endmembers in a pixel is

$$\mathbf{x} = \mathbf{M}\mathbf{a} + \mathbf{w} \tag{2.15}$$

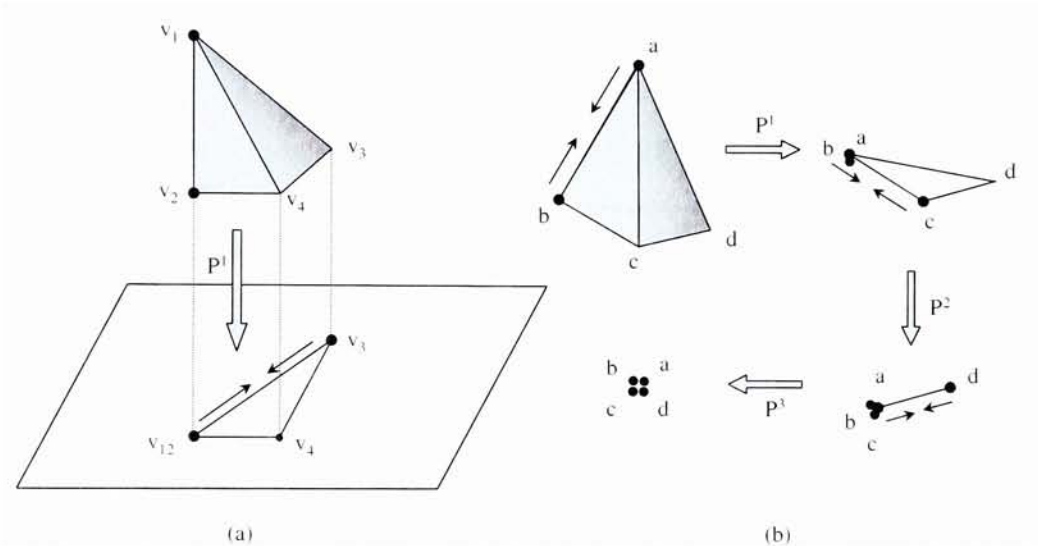


Figure 2.5: Illustration of (a) the preservation of vertices of a simplex through projection of a data set onto the difference in two vertices of a simplex and (b) the concept of maximum distance determination and sequential projection to find the vertices of a simplex spanning the data space.

where $\mathbf{x}$ is the spectrum of the mixed pixel, $\mathbf{M}$ is a matrix of endmembers, $\mathbf{a}$ is a vector of endmember weights, and $\mathbf{w}$ is a vector of residuals to account for noise and lack of fit. The LMM can be constrained both partially and fully. If we enforce an additivity constraint such that

$$\sum_{i=1}^{N} a_i = 1 \tag{2.16}$$

then we partially constrain the model, where N is the number of endmembers. If we also enforce the positivity constraint that $a_i \geq 0$, then we have fully constrained the model. If the endmember spectra are randomly and independently drawn from multivariate normal distribution, we have the stochastic mixing model (SMM) described in Sec. 2.2.2. The LMM and its relationship to geometry can be found in Appendix D.

2.3 Approaches to Target Detection

In a general sense, the key factors that determine the taxonomy of a hyperspectral detection algorithm are: *(i)* the type of model used for spectral (target or background) variability, *(ii)* the composition of the pixel under test (pure or mixed), and *(iii)* the model used to describe mixed pixels [34]. Up to this point we have discussed all of the above mentioned factors except for how they are utilized in a hyperspectral target detection scheme. In this section, we describe some of the algorithms used in full and sub-pixel target detection.

2.3.1 Full-Pixel Target Detection Algorithms

The vernacular "fully resolved" means that the pixel is *completely filled* by the material of interest. In detection, this means that the pixel either contains target $\mathbf{t}$ or background $\mathbf{b}$, not both. The only other effects inherent in both cases are factors such as secondary illumination, shading and sensor noise. Therefore, the model and hypothesis we assume is

$$\begin{aligned} \mathcal{H}_0\text{:} \quad & \mathbf{x} \in \mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b) \\ \mathcal{H}_1\text{:} \quad & \mathbf{x} \in \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t) \end{aligned}$$

where in the stochastic representation the assumption is that $\mathbf{x}$ comes from either the multivariate distribution $\mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ or $\mathcal{N}(\boldsymbol{\mu}_b, \boldsymbol{\Sigma}_b)$.

Background and Target Statistics Known

If we proceed with a likelihood function, as described previously in Sec. 2.1.1, and the normal assumption then the LR can be described as

$$\mathcal{L}(\mathbf{x}) = \frac{p(\mathbf{x};\mathcal{H}_1)}{p(\mathbf{x};\mathcal{H}_0)} = \frac{(2\pi)^{-p/2}|\boldsymbol{\Sigma}_t|^{-1/2}\exp\left[-0.5(\mathbf{x}-\boldsymbol{\mu}_t)^T\boldsymbol{\Sigma}_t^{-1}(\mathbf{x}-\boldsymbol{\mu}_t)\right]}{(2\pi)^{-p/2}|\boldsymbol{\Sigma}_b|^{-1/2}\exp\left[-0.5(\mathbf{x}-\boldsymbol{\mu}_b)^T\boldsymbol{\Sigma}_b^{-1}(\mathbf{x}-\boldsymbol{\mu}_b)\right]} \tag{2.17}$$

which reduces to

$$\mathcal{L}(\mathbf{x}) = \frac{|\boldsymbol{\Sigma}_b|^{1/2}}{|\boldsymbol{\Sigma}_t|^{1/2}}\exp\left[0.5(\mathbf{x}-\boldsymbol{\mu}_b)^T\boldsymbol{\Sigma}_b^{-1}(\mathbf{x}-\boldsymbol{\mu}_b) - 0.5(\mathbf{x}-\boldsymbol{\mu}_t)^T\boldsymbol{\Sigma}_t^{-1}(\mathbf{x}-\boldsymbol{\mu}_t)\right]. \tag{2.18}$$

The quadratic term $(\mathbf{x}-\boldsymbol{\mu}_y)^T\boldsymbol{\Sigma}_y^{-1}(\mathbf{x}-\boldsymbol{\mu}_y)$, for y equal to t or b, is a statistical distance measure and is referred to as the square of the *Mahalanobis distance*. If the target and background covariances are whitened, (*i.e.*, $\boldsymbol{\Sigma} = \sigma^2\mathbf{I}$) then the distance measure becomes a simple Euclidean one. Taking the natural logarithm (a monotonically

increasing transformation) of both sides of Eq. (2.18) does not change the relation. Therefore, we have the test statistic (T)

$$T(\mathbf{x}) = \ln \mathcal{L}(\mathbf{x}) = \ln \left(\frac{|\Sigma_b|}{|\Sigma_t|} \right)^{1/2} + 0.5(\mathbf{x}-\boldsymbol{\mu}_b)^T \Sigma_b^{-1} (\mathbf{x}-\boldsymbol{\mu}_b) - 0.5(\mathbf{x}-\boldsymbol{\mu}_t)^T \Sigma_t^{-1} (\mathbf{x}-\boldsymbol{\mu}_t). \tag{2.19}$$

For every test pixel $\mathbf{x}$, the background and target covariances will usually remain constant unless they are determined through other means (*cf.* Sec. 2.2.1). Furthermore, the natural logarithm of the ratio of the covariance determinants will be constant for each pixel under test. This constant does not effect our discriminant and is usually ignored. In doing so, we have

$$T_{QD}(\mathbf{x}) = 0.5(\mathbf{x} - \boldsymbol{\mu}_b)^T \Sigma_b^{-1} (\mathbf{x} - \boldsymbol{\mu}_b) - 0.5(\mathbf{x} - \boldsymbol{\mu}_t)^T \Sigma_t^{-1} (\mathbf{x} - \boldsymbol{\mu}_t). \tag{2.20}$$

This is called a *quadratic detector*. It simply compares the distances of the observed spectrum $\mathbf{x}$ from the centers of the two target and background classes.

In the event that the target and background classes have the same (or pooled) covariance, $\Sigma_t = \Sigma_b = \Sigma$, Eq. (2.19) becomes linear in form and reduces to

$$T_{MF}(\mathbf{x}) = (\boldsymbol{\mu}_t - \boldsymbol{\mu}_b)^T \Sigma^{-1} \mathbf{x}. \tag{2.21}$$

This detector is known as the *matched filter* and referred to in the statistics community as *Fisher's linear discriminant* [20]. If we apply the matched filter to mean-centered data and replace the mean of the target distribution, $\boldsymbol{\mu}_t$ with a target vector, $\mathbf{t}$ we have

$$T_{MF\_mc}(\mathbf{x}) = (\mathbf{t} - \boldsymbol{\mu}_b)^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}_b). \tag{2.22}$$

Finally we can normalize Eq. (2.22) with the variance adjusted magnitude of the target so as to produce a scalar quantity with a maximum value of $T(\mathbf{x}) = 1$

(when the data is the target). This is analogous to an "abundance" of how much target there is in a pixel and can be expressed as

$$T_{MF\_norm}(\mathbf{x}) = \frac{(\mathbf{t}-\boldsymbol{\mu}_b)^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_b)}{(\mathbf{t}-\boldsymbol{\mu}_b)^T\boldsymbol{\Sigma}^{-1}(\mathbf{t}-\boldsymbol{\mu}_b)}. \tag{2.23}$$

Background Statistics Known, Target Mean Unknown

Now we present the case where the background statistics are known but the information about the target mean is unknown. Here, we need to alter our detection statistic because no *a priori* information about the target class is available. If we strictly base the test statistic on $\boldsymbol{\mu}_b$ while assuming $\boldsymbol{\Sigma}_t = \boldsymbol{\Sigma}_b = \boldsymbol{\Sigma}$, then we have

$$T_{AD}(\mathbf{x}) = (\mathbf{x}-\boldsymbol{\mu}_b)^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_b) \tag{2.24}$$

which is the statistical distance of the test pixel from the mean of the background class $\boldsymbol{\mu}_b$. This is known as an *anomaly detector* [33] and is widely used, along with its variants, on hyperspectral data to find anomalous targets.

Target Mean Known (Geometric Method)

Lastly, we introduce a well known (full pixel) algorithm that is background independent. That is, no information about the background is required in the algorithm only the target needs to be known. The algorithm simply computes angles formed by a target and image pixels, and is appropriately called the *spectral angle mapper* (SAM) [29]. The spectral angle θ in the SAM algorithm is computed as a variant of the scalar product in $\mathbb{R}^n$ space. If $\mathbf{x}$ and $\mathbf{t}$ are two nonzero vectors we have

$$\mathbf{x}^T\mathbf{t} = \|\mathbf{x}\|\ \|\mathbf{t}\|\cos\theta \tag{2.25}$$

where the spectral angle is computed as

$$T_{SAM}(\mathbf{x}) = \theta = \arccos \frac{\mathbf{t}^T \mathbf{x}}{\|\mathbf{t}\| \ \|\mathbf{x}\|} \tag{2.26}$$

where $\mathbf{t}$ is the target vector. Though this algorithm is widely used, it finds a deficiency in its inability to model a target spectrum under various illumination conditions. To first order, the algorithm is invariant to vector magnitude changes because we are measuring an angle. However, in scenarios where the spectral "color" changes due to illumination variations (*i.e.*, more downwelled radiance) this approach will fail. An improved method that address this illumination issue will be discussed in Sec. 2.4.

2.3.2 Sub-Pixel Target Detection Algorithms

In contrast to a fully resolved target pixel, a sub-pixel target will only fill a portion of the pixel of interest. The regions in the pixel that are not target are considered background. Since we have a "mixed" pixel, it would behoove us to use a model that accounts for this mixture. More often than not, a standard linear mixing model (and its derivatives) is used for this purpose (*cf.* Sec. 2.2.4).

The algorithms that are developed for the sub-pixel cases are completely dependent on how the target and background spaces are modeled (geometric versus statistical). For example, if we characterize the background as *structured*, we use a geometric sub-space or a linear mixing model. On the other hand, if we treat the background as being *unstructured*, then we use a family of algorithms that are based on statistics. The target space is usually characterized using subspace models, which will be assumed throughout this section, though we will be investigating other approaches on modeling target spaces in this research.

In this sub-section, we will first explore some of the structured (geometric) models followed by an analysis of some statistical unstructured models. All of these algorithms find their roots in signal processing which falls in the domain of electrical engineering and not so much imaging science. Since much of the descriptive work, including comparative and overview studies, has been published extensively throughout the literature [32–35], we will only state the algorithms and defer the details via the appropriate reference.

Structured (Geometric) Background Models

Let us consider an image consisting of N pixels. Each pixel is represented by a p-dimensional vector $\mathbf{x}_i$, where p is the number of spectral bands and $i = 1, 2, \ldots, N$. We then assume the following structured linear mixing model (LMM)

$$\mathbf{x}_i = \mathbf{T}\mathbf{a}_i + \mathbf{B}\mathbf{b}_i + \boldsymbol{\epsilon}_i \tag{2.27}$$

where $\mathbf{T}$ is a matrix of target endmembers, $\mathbf{B}$ is a matrix of background endmembers, and $\mathbf{a}_i$ and $\mathbf{b}_i$ are unknown weighting vectors to be estimated in a least-squares sense. The vector $\boldsymbol{\epsilon}_i$ represents approximation errors or lack of fit. This can be due to noise in the data or modeling error (or both). Finally, we assume $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$, though this is not typically true when using real data [36]. If we focus our discussion on a single pixel and omit the subscript i, then the competing hypotheses will be

$$\begin{aligned} \mathcal{H}_0&: \quad \mathbf{x} = \mathbf{B}\mathbf{b} + \boldsymbol{\epsilon} \\ \mathcal{H}_1&: \quad \mathbf{x} = \mathbf{T}\mathbf{a} + \mathbf{B}\mathbf{b} + \boldsymbol{\epsilon} \end{aligned}$$

Generating a generalized likelihood ratio (GLR) (*cf.* Sec. 2.1.3) produces the following detector or test statistic [36]

$$T_{GLR}(\mathbf{x}) = \left(\frac{\mathbf{x}^T\mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x}}{\mathbf{x}^T\mathbf{P}_{\mathbf{Z}}^{\perp}\mathbf{x}}\right)^{p/2} \tag{2.28}$$

where $\mathbf{Z} = [\mathbf{T}, \mathbf{B}]$ is a matrix consisting of all columns of $\mathbf{T}$ and $\mathbf{B}$, and $\mathbf{P}_{\mathbf{Y}}^{\perp}$ (for $\mathbf{Y}$ equal to $\mathbf{B}$ or $\mathbf{Z}$) is the matrix of the projection onto the space orthogonal to the space generated by columns of $\mathbf{Y}$. That is,

$$\mathbf{P}_{\mathbf{Y}}^{\perp} = \mathbf{I} - \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T \tag{2.29}$$

$$= \mathbf{I} - \mathbf{Y}\mathbf{Y}^{\dagger} \tag{2.30}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{Y}^{\dagger} = (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T$ is the pseudo-inverse of $\mathbf{Y}$. It has been shown that T_{GLR} is a monotonic function of a match subspace detector (MSD) [42]

$$T_{MSD}(\mathbf{x}) = \frac{\mathbf{x}^T(\mathbf{P}_{\mathbf{B}}^{\perp} - \mathbf{P}_{\mathbf{Z}}^{\perp})\mathbf{x}}{\mathbf{x}^T\mathbf{P}_{\mathbf{Z}}^{\perp}\mathbf{x}}. \tag{2.31}$$

More specifically,

$$T_{GLR}(\mathbf{x}) = (T_{MSD}(\mathbf{x}) + 1)^{p/2}. \tag{2.32}$$

That is, the MSD is equivalent to the GLR and is often referred to as the adaptive subspace detector (ASD) $T_{ASD}(\mathbf{x})$, [34,35]. In statistics, the quantity $T_{MSD}(\mathbf{x})$ (and $T_{ASD}(\mathbf{x})$) is denoted by $F(\mathbf{x})$ and is known as the *F-test* [34].

Another widely used sub-pixel detection algorithm that uses the orthogonal projection operator $\mathbf{P}_{\mathbf{B}}^{\perp}$ is called the orthogonal subspace projector (OSP) [13] and is described as

$$T_{OSP}(\mathbf{x}) = \frac{\mathbf{t}^T\mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x}}{\mathbf{t}^T\mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{t}} \tag{2.33}$$

where $\mathbf{t}$ is a target vector. With the normalization, this test statistic produces an estimate of the target abundance similar to Eq. (2.23) except we are now using the projection operator $\mathbf{P}_{\mathbf{B}}^{\perp}$ in place of the covariance $\mathbf{\Sigma}$.

We can generate a background independent detector similar to the spectral angle mapper (SAM) (described in Sec. 2.3.1) by ignoring the background such that $\mathbf{B} = \mathbf{0}$ in Eq. (2.31). That is $\mathbf{P}_{\mathbf{B}}^{\perp} \Rightarrow \mathbf{I}$ and $\mathbf{P}_{\mathbf{Z}}^{\perp} \Rightarrow \mathbf{P}_{\mathbf{T}}^{\perp}$. This algorithm, called the matched filter for targets in white noise (MFWN) [35], is described as

$$T_{MFWN}(\mathbf{x}) = \frac{\mathbf{x}^T(\mathbf{I} - \mathbf{P}_{\mathbf{T}}^{\perp})\mathbf{x}}{\mathbf{x}^T\mathbf{P}_{\mathbf{T}}^{\perp}\mathbf{x}} = \frac{\mathbf{x}^T\mathbf{P}_{\mathbf{T}}\mathbf{x}}{\mathbf{x}^T\mathbf{P}_{\mathbf{T}}^{\perp}\mathbf{x}} \tag{2.34}$$

where

$$\mathbf{P}_{\mathbf{T}} = \mathbf{T}(\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T \tag{2.35}$$

and

$$\mathbf{P}_{\mathbf{T}}^{\perp} = \mathbf{I} - \mathbf{T}(\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T \tag{2.36}$$

with $\mathbf{T}$ being a matrix of target endmembers. Equation (2.34) is related to the SAM algorithm by

$$T_{MFWN}(\mathbf{x}) = \frac{T_{SAM}^2(\mathbf{x})}{1 - T_{SAM}^2(\mathbf{x})}. \tag{2.37}$$

Unstructured (Stochastic) Background Models

We now focus on modeling the background in a stochastic sense. Here, we think of finding a mean centered target in additive background noise $\boldsymbol{\epsilon}_b \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_b)$. Therefore, the model we assume is

$$\begin{aligned} \mathcal{H}_0&: \quad \mathbf{x} = \boldsymbol{\epsilon}_b \\ \mathcal{H}_1&: \quad \mathbf{x} = \mathbf{T}\mathbf{a} + \boldsymbol{\epsilon}_b \end{aligned}$$

where $\mathbf{x}$ either comes from $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_b)$ under $\mathcal{H}_0$ or $\mathcal{N}(\mathbf{Ta}, \mathbf{\Sigma}_b)$ under $\mathcal{H}_1$.

Using a generalized likelihood ratio approach (GLR), we obtain the following detector

$$T_{GLR}(\mathbf{x}) = \frac{\mathbf{x}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{T}(\mathbf{T}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{T})^{-1}\mathbf{T}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{x}}{N + \mathbf{x}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{x}} \tag{2.38}$$

where $\hat{\mathbf{\Sigma}}_b$ is the MLE of the covariance matrix and N is the number of background pixels. This detector was first developed by Kelly [24] and is sometimes referred to as the Kelly detector T_{Kelly}. In the event that the target is deterministic (*i.e.*, no spectral variability), the dimensionality of $\mathbf{T}$ is reduced to a single vector $\mathbf{t}$ which yields

$$T_{GLR}(\mathbf{x}) = \frac{(\mathbf{t}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{x})^2}{(\mathbf{t}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{t})(1 + \mathbf{x}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{x})}. \tag{2.39}$$

Due to filling factors caused by the amount of target in the pixel, the background covariance in both hypotheses may not be the same. To account for this we can introduce a variance scaling factor σ to our model. This produces

$$\begin{aligned} \mathcal{H}_0\text{:} \quad & \mathbf{x} = \boldsymbol{\epsilon}_b \\ \mathcal{H}_1\text{:} \quad & \mathbf{x} = \mathbf{Ta} + \sigma\boldsymbol{\epsilon}_b \end{aligned}$$

which says under the alternative hypothesis $\mathcal{H}_1$ that $\mathbf{x} \sim \mathcal{N}(\mathbf{Ta}, \sigma^2\mathbf{\Sigma}_b)$. Using this altered model and a GLR approach leads to the adaptive coherence estimator (ACE) [27, 28]

$$T_{ACE}(\mathbf{x}) = \frac{\mathbf{x}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{T}(\mathbf{T}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{T})^{-1}\mathbf{T}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{x}}{\mathbf{x}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{x}}. \tag{2.40}$$

Using the same assumption about deterministic targets, Eq. (2.40) reduces to

$$T_{ACE}(\mathbf{x}) = \frac{(\mathbf{t}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{x})^2}{(\mathbf{t}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{t})(\mathbf{x}^T\hat{\mathbf{\Sigma}}_b^{-1}\mathbf{x})}. \tag{2.41}$$

If we remove the normalization by the variance scaled target pixel $\mathbf{x}$, we have the adaptive matched filter (AMF) [41] which is express as

$$T_{AMF}(\mathbf{x}) = \frac{(\mathbf{t}^T \hat{\mathbf{\Sigma}}_b^{-1} \mathbf{x})^2}{\mathbf{t}^T \hat{\mathbf{\Sigma}}_b^{-1} \mathbf{t}}. \tag{2.42}$$

Both the AMF and ACE detectors are special cases of the GLR detector in Eq. (2.38).

Another detection algorithm that is analogous to the OSP detector in Eq. (2.33) is the constrained energy minimization (CEM) [9] algorithm. The CEM method constructs a linear operator that minimizes the response of the unknown background signatures while the response of the signatures of interest is constrained to a desired constant level. This is accomplished by first minimizing the total output energy, E of the linear operator $\mathbf{Y}$. That is

$$E = \sum_{i=0}^{N} \left(\mathbf{Y}^T \mathbf{x}_i\right)^2 \tag{2.43}$$

where $\mathbf{x}_i$ is the i^{th} image pixel and N is the total number of pixels. This is subject to the unity constraint $\mathbf{Y}^T\mathbf{t} = 1$. The constrained minimization problem is solved using the method of Lagrange multipliers defined as

$$h(\mathbf{Y}, \gamma) = \sum_{i=0}^{N} \left(\mathbf{Y}^T \mathbf{x}_i\right)^2 - \gamma \left(\mathbf{Y}^T \mathbf{t} - 1\right). \tag{2.44}$$

The solution is obtained by calculating the variable γ that minimizes the Lagrange function $h(\mathbf{Y}, \gamma)$. Thus the solution is

$$\mathbf{Y} = \frac{\mathbf{\Sigma}^{-1}\mathbf{t}}{\mathbf{t}^T\mathbf{\Sigma}^{-1}\mathbf{t}} \tag{2.45}$$

which produces the test statistic

$$T_{CEM}(\mathbf{x}) = \frac{\mathbf{t}^T\boldsymbol{\Sigma}^{-1}\mathbf{x}}{\mathbf{t}^T\boldsymbol{\Sigma}^{-1}\mathbf{t}} \tag{2.46}$$

where $\boldsymbol{\Sigma}$ is the background covariance. The CEM does not perform well in the presence of low probability background signatures and is very sensitive to noise [47]. We can see that the CEM algorithm resembles that of the standard matched filter (MF) of Eq. (2.23) discussed earlier.

2.4 Target Spaces and Physics Based Modeling (PBM)

Thus far we have introduce a host of models and algorithms used for target detection that may or may not require target information. The underlying assumption in many of these models is that the data has already been atmospherically compensated. Furthermore, many of the (non-anomaly) algorithms assume the target is deterministic and does not vary. In the event that one wishes to include target variability, it is usually in the form of variation due to image or sensor noise.

In this section we illustrate a method, based on a physics model, that generates a sensor reaching target *space*, with variation due to physical effects, which does not require image data to be atmospherically compensated.

2.4.1 The Invariant Algorithm: An Application of PBM

Many algorithms attempt to account for atmospheric and illumination effects using methods to invert to reflectance and then perform the target search in the reflectance domain. Another approach, first developed by Healey and Slater [14], takes the opposite approach and attempts to transform a target spectrum into the image domain in such a fashion that atmospheric and illumination variations are normalized out

of the detection process.

Even if an aerial sensor platform has spectra that are radiometrically calibrated, the identification of materials is complicated by spatial and temporal variation in illumination as well as atmospheric conditions. Atmospheric compensation algorithms, sometimes with the aid of ground truth, help in relating surface reflectance to sensor radiance. However, they typically assume that direct solar radiation is the dominant contributor to ground-surface illumination [14].

As for target detection itself, algorithms exists that rely only on a single spectral vector of the material of interests. These approaches work well as long as the dimensionality of the sensor data exceeds the dimensionality of the "set" of spectral radiance vectors that can be obtained for a single material as conditions change [14]. The invariant approach uses a physical model as well as an atmospheric propagation model to vary atmospheric and illumination conditions. From this, a reduced set of observed spectral radiance vectors is found in conjunction with a maximum likelihood algorithm to identify materials, invariant of illumination and atmospheric effects.

2.4.2 An Illumination Variant Example

An example of where traditional, illumination variant, detection algorithms might fail is when the target appears both in direct (exposed) and indirect (concealed) sunlight. Figure 2.6 shows a spectrum from the HYDICE sensor of a green cotton fabric target illuminated by both direct and indirect sunlight.

Here, the vectors are Euclidean normalized as

$$\mathbf{L}_{norm} = \frac{\mathbf{L}}{\|\mathbf{L}\|} \tag{2.47}$$

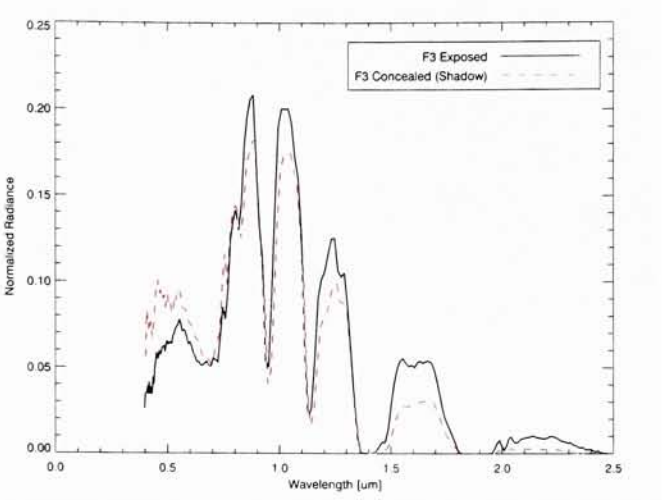


Figure 2.6: Normalized HYDICE spectra for panel made of green cotton fabric. The solid line represents the spectral radiance of the panel in direct sunlight while the dotted red line shows illumination effects when the panel was in the shade.

where $\mathbf{L}$ is the radiance vector. The Euclidean length or magnitude is defined in terms of the scalar product, $\|\mathbf{L}\| = (\mathbf{L}^T\mathbf{L})^{1/2}$. It is apparent that the dotted line contains more blue energy than the solid line. This corresponds to the panel that was in the shade thus receiving the bulk of its illumination from downwelled radiance. Here a simple automated algorithm like the spectral angle mapper (SAM) (*cf.* Sec. 2.3.1), which simply computes angles formed by a target and image pixels, will perform poorly because of its inability to model the target spectrum under various illumination conditions.

If we take the target vector $\mathbf{t}$, as measured in direct (exposed) illumination, and use it in the SAM algorithm to search for the panel in the image where the panel was measured in the shade ($\mathbf{x}_{i,j}$), we get the results seen in Figure 2.7 where the spectral angle mapper is expressed as

Figure 2.7: False alarms (yellow pixels) when finding the F3 spectrum as measured in direct sunlight in an image where the (same) panel is placed in the shade. The SAM algorithm was used (in radiance space) with a threshold of 0.26 radians. The white circle area refers to the location of the F3 target in the shade. Notice no detects in this region.

$$\theta = \arccos \frac{\mathbf{t}^T \mathbf{x}_{i,j}}{\|\mathbf{t}\| \ \|\mathbf{x}_{i,j}\|}. \tag{2.48}$$

We can see that there are many false alarms (yellow pixels) at the threshold level shown (0.26 radians) with no correct detects. This clearly shows that the approach needs to somehow compensate for atmospheric and illumination conditions.

Physics-Based Modelling of Spectra

The invariant approach attempts to account for illumination and atmospheric effects by generating permutations of how the target spectrum might appear based on a given illumination condition and atmospheric condition. One way to do this is to generate a target-radiance model which incorporates the above mentioned parameters. Once this is established, the parameters can be varied in such a way so as to

generate a multitude of possible, physically realizable, target-radiance spectra.

The physics-based governing equation which generates a sensor reaching radiance L_s is one used by Schott [46] and is given by

$$L_s(\lambda) = \tau_2(\lambda)r(\lambda)\left[\tau_1(\lambda)\frac{E_s^{'}(\lambda)\cos\sigma^{'}}{\pi} + \int_{\phi=0}^{2\pi}\int_{\theta=0}^{\pi/2} L_d(\theta,\phi,\lambda)\cos\sigma\ \sin\sigma\ \mathrm{d}\sigma\ \mathrm{d}\phi\right] + L_u(\lambda) \qquad (2.49)$$

where $\tau_2(\lambda)$ is the transmission along the target-sensor path, $r(\lambda)$ is the spectral reflectance of the matte surface, $\tau_1(\lambda)$ is the transmission along the sun-target path, $E_s^{'}(\lambda)$ is the exoatmospheric irradiance, $\sigma^{'}$ is the solar zenith angle, L_d is the downwelled radiance, and L_u is the upwelling radiance.

Of all the parameters mentioned in the physical model, only the spectral reflectance r is directly related to the target. The other parameters tend to modulate the target signature before it reaches the sensor through absorption and scattering processes. The more prominent atmospheric gases include water vapor (H_2O), ozone (O_3), oxygen (O_2), methane (CH_4), nitrous oxide (N_2O), carbon monoxide (CO), and carbon dioxide (CO_2). To generate the spectral target permutations an atmospheric propagation model (MODTRAN) is used. This radiative transfer code is used to characterize the effect of atmosphere and illumination on radiance for each reflectance vector considered.

In the case of Healy and Slater [14], their work generated a total of 28 672 unique conditions, of which 17 920 were physically realizable (*i.e.*, the relative humidity did not exceed 100%). This means that for each reflectance vector $\mathbf{r}$, 17 920 estimates of how the target might appear at the sensor in radiance space were generated.

Reduction of Dimensionality

Since the idea of the invariant technique is to account for all possible manifestations of a target-spectrum, one runs the risk of generating significantly large data sets that then have to be used in some type of detection scheme. These large data sets could put a burden on the target detection algorithm in terms of computational feasibility. We therefore look to work in a dimensionally reduced data space.

We wish to generate a target *subspace* from the original space $\mathbb{R}^n$. In doing so, we first set out to approximate the target-spectra representation using a linear representation. If the linear representation is valid, we can then make approximations through use of a potentially smaller (basis) vector set. To do this, a set of orthonormal basis vectors is found using the singular value decomposition (SVD)(*cf.* Sec. 2.2.3) method to solve for the number of basis vectors that minimize a sum-squared error.

We first wish to establish a linear model that approximates a given radiance spectrum, L_i. The linear approximation $\hat{L}_i$ is expressed as

$$\hat{L}_i = \sum_{j=1}^{N} a_{ij}\mathbf{m}_j \quad 1 \leq i \leq C \tag{2.50}$$

where we have N basis vectors and C estimates of radiance produced for a single reflectance vector (*e.g.*, Healy and Slater generated $C = 17\,920$). Assuming there are p-wavelength centers, the vectors $\mathbf{m}_j = [m_j(\lambda_1), m_j(\lambda_2), \cdots, m_j(\lambda_p)]^T$, for $1 \leq j \leq N$, define a fixed orthonormal basis for the material. The constants $a_{i,j}$ are weighting coefficients.

In matrix form Eq. (2.50) is express as

$$\hat{L}_i = \mathbf{M}\mathbf{a}_i = [\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_N]\,\mathbf{a}_i \tag{2.51}$$

where $\mathbf{M}$ is a (p-band$\times N$-basis vector) matrix of N column vectors $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_N$ and $\mathbf{a}_i$ is an N element column vector made up of the weights used for the i^{th} treatment to estimate the i^{th} radiance vector, L_i. This is our linear mixing model (LMM) described in Sec. 2.2.4. The weights for a particular radiance vector can be solved by standard least-squares methods (for $p \geq N$). That is

$$\mathbf{a}_i = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T L_i. \tag{2.52}$$

The accuracy of the linear approximation for a single L_i and N-basis vector set is defined by the squared error (SE) as

$$\mathrm{SE}_i(N) = \left\| L_i - \hat{L}_i \right\|^2 = \left\| L_i - \sum_{j=1}^{N} a_{i,j}\mathbf{m}_j \right\|^2 = \|L_i - \mathbf{M}\mathbf{a}_i\|^2 . \tag{2.53}$$

Similarly, a total sum squared error (SSE) value can be computed for all instances of L_i by summing up the individual squared errors for a given value of N. That is,

$$\mathrm{SSE}(N) = \sum_{i=1}^{C} \mathrm{SE}_i(N). \tag{2.54}$$

Again the singular value decomposition (SVD) is used to compute the set of orthonormal basis vectors (*i.e.*, $\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_N$) that minimizes SSE($N$).

Lastly, an average normalized error, as a function of N, is computed as

$$\bar{E}_T(N) = \frac{\mathrm{SSE}(N)}{\mathrm{SSL}} = \frac{\mathrm{SSE}(N)}{\sum_{i=1}^{C} \|L_i\|^2} \tag{2.55}$$

where SSL is the sum-squared radiance.

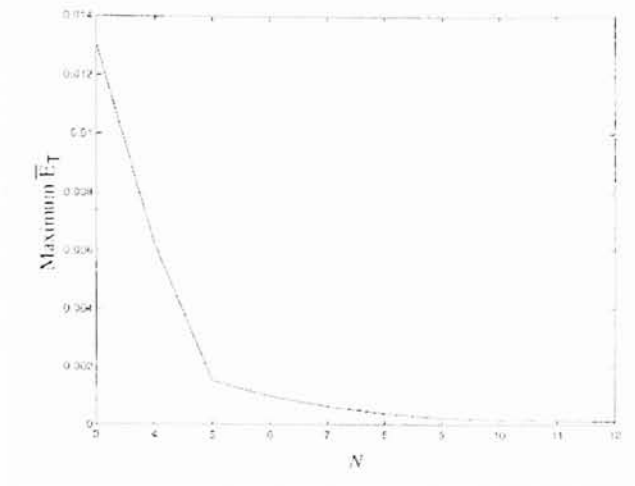


Figure 2.8: Average error as a function of number of basis vectors [14].

To get an estimate of how many basis vectors might be used for a given material, Healy and Slater generated $C = 17\,920$ observations for each of 498 materials in the USGS spectral reflectance database. For each material they generated N basis vectors ranging from 3 to 12. This range agrees with geometric basis vector selection studies performed by Bajorski and Ientilucci [2, 3]. The average error $\bar{E}_T(N)$ for each material was found and the maximum for each value of N retained and plotted against N to estimate the maximum number of dimensions needed to characterize the radiance space (see Figure 2.8). From this it was determined that a 9 dimensional space (*i.e.*, 9 basis vectors) could adequately represent each of the 17 920 spectral vectors. Thus reducing the dimensionality of the data set from 210 (number of bands in the HYDICE sensor) to 9. Additionally, it is noted that each of the 9-basis vectors sets is different for each material. This result tells us that the "target spectrum" can be readily searched for using a smaller 9-dimensional space instead of the original 210 dimensions.

Maximum-Likelihood Classification

We can now implement the reduced data in a classification scheme for material identification. This is done by first considering *normalized* spectral vectors $\mathbf{L}_{norm} = \mathbf{L}/\|\mathbf{L}\|$, as in Eq. (2.47). Normalization avoids scaling errors due to the magnitude of the vector. That is, we normalize by the overall brightness. We can then express a single observation of a material as

$$\mathbf{L}_{norm} = \sum_{j=1}^{N} a_j \mathbf{m}_j + \boldsymbol{\epsilon} \tag{2.56}$$

where the coefficients a_j depend on the condition under which the sample is viewed and $\boldsymbol{\epsilon}$ is our p-dimensional residual error vector. We choose N so that the approximation error is small and model $\boldsymbol{\epsilon}$ as being Gaussian, having zero mean with small covariance values.

The probability of a vector $\mathbf{L}_{norm}$ occurring given a reflectance spectrum $\mathbf{r}$ and a weighting coefficient vector $\mathbf{a}$ can be expressed as

$$P(\mathbf{L}_{norm}|\mathbf{r},\mathbf{a}) = \frac{1}{(2\pi)^{p/2}\,|\boldsymbol{\Sigma}|^{1/2}} \exp(-0.5\,\mathbf{k}^T\,\boldsymbol{\Sigma}^{-1}\,\mathbf{k}) \tag{2.57}$$

where $\boldsymbol{\Sigma}$ is the covariance matrix of the residual error $\boldsymbol{\epsilon}$. We let $\mathbf{k}$ be a vector that is a specific estimate of the residual error and is expressed as

$$\mathbf{k} = \mathbf{L}_{norm} - \sum_{j=1}^{N} a_j \mathbf{m}_j \quad = \mathbf{L}_{norm} - \mathbf{M}\mathbf{a}. \tag{2.58}$$

We recognize that the information about $\mathbf{r}$ in this process is contained in the basis vectors that make up $\mathbf{M}$ while the specifics of the atmospheric and illumination conditions are contained in the coefficient weighting vector $\mathbf{a}$.

If $\mathbf{\Sigma}$ is known (in most cases it is not) then the maximum likelihood values for the a_j coefficients can be obtained by differentiating $\mathbf{k}^T\mathbf{\Sigma}\mathbf{k}$ with respect to each coefficient, setting the results to zero, and solving for the a_j values.

For example, if we assume the case where the elements of the residual error vector $\boldsymbol{\epsilon}$ are independent and identically distributed (*i.e.*, the same variance or i.i.d.), then the maximum likelihood estimates $\hat{\mathbf{a}}$ are given by

$$\not{\epsilon}^0 = \mathbf{L}_{norm} - \mathbf{M}\hat{\mathbf{a}} \tag{2.59}$$

where $\hat{\mathbf{a}}$ can be computed as

$$\hat{\mathbf{a}} = \mathbf{M}^{-1}\mathbf{L}_{norm} \tag{2.60}$$

and $\mathbf{M}^{-1} = \mathbf{M}^T$ since the vectors are orthonormal. Using this property Eq. (2.60) can also be written in familiar form as

$$\hat{\mathbf{a}} = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{L}_{norm} \tag{2.61}$$

which solves for the weighting vector in a least-squares senses and takes advantage of the Moore-Penrose pseudoinverse. This result naturally minimizes the error

$$\|\mathbf{L}_{norm} - \mathbf{M}\mathbf{a}\|. \tag{2.62}$$

In practice, the weighting vector $\hat{\mathbf{a}}$ is solved for each pixel and then used in Eq. (2.57) to solve for the likelihood of observing the vector $\mathbf{L}_{norm}$, assuming the pixel is generated by imaging a material with a spectral reflectance characterized by the vector $\mathbf{r}$ which is used to generate the basis vectors that make up matrix $\mathbf{M}$. The result is a probability-like image that can be thresholded and used for target identification.

We also recognize that for the case of i.i.d. residuals, the probability function in Eq. (2.57) is a monotonically decreasing function of the magnitude of the error

$$\|\mathbf{L}_{norm} - \mathbf{M}\hat{\mathbf{a}}\| \tag{2.63}$$

where $\hat{\mathbf{a}}$ is computed using Eq. (2.60) or (2.61). Here, the error term alone can be thresholded and used for material identification.

Healy and Slater [14] further demonstrate this method of target detection by comparing it to the spectral angle mapper (SAM) algorithm. Results showed that the SAM algorithm incurred many false alarms while the invariant algorithm excelled when a target panel of interest was placed in the shade. However, this approach to target identification assumes you know the spectral reflectance of each target material you are looking for. This information is needed in order to create all the permutations (and eventually a sub-space) of how the target might appear at the sensor. Furthermore, the presented method only addresses the issues of target identification for fully resolved target pixels.

2.4.3 Target Space Considerations: Clouds

This section briefly introduces some spectral modeling effects, due to cloud cover, that could potentially affect the target sub-space modeling process. In an ideal situation, imagery is captured free of cloud cover. However, in some situations, this may not be the case. Cloud cover not only reduces the overall *magnitude* of direct solar radiation onto the target, but changes its spectral character as well. This can be seen in Figure 2.9. Here we have an experiment [4] where a spectrometer was pointed in the direction of the Sun, absent of any clouds (black line). This was followed by subsequent measurements in which a cloud passed in front of the Sun.

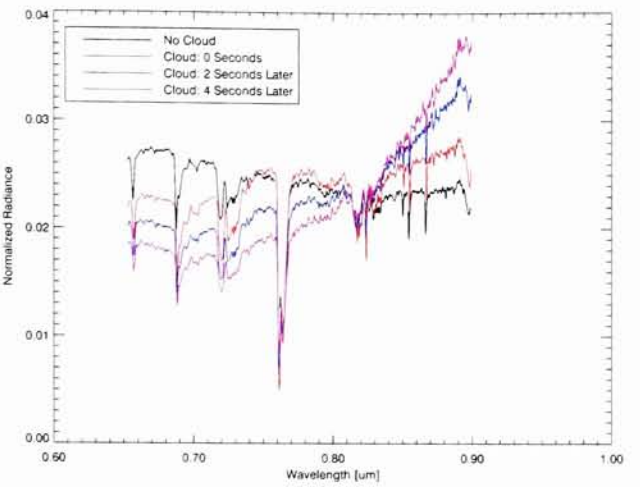


Figure 2.9: Experiment in which a spectrometer is pointed in the direction of the Sun. Black line is the normalized radiance in the absence of a cloud while subsequent measurements are representative of a cloud passing in front of the Sun.

Intuitively we know that the overall magnitude is decreased but it isn't until we normalize the spectra that we see changes in spectral character.

As a precursor, the imagery collected in this research was free of clouds. Therefore, this phenomena was not taken into account or modeled. However, it should be noted that this source of error can play an important role in the target space generation process, though its impact on target detection has yet to be determined and maybe the subject of future work.

2.5 Hybrid Approaches

Thus far we have illustrated traditional detectors that were developed based on either geometric or statistical concepts. In this section we introduce "hybrid" detec-

tion approaches that take advantage of both mathematical concepts. For example, an algorithm may model the background with first and second order statistics as well as use a (geometric) projection operator to manipulate the data for additional interpretation. We first introduce the Finite Target Matched Filter (FTMF) which leads us into a derivation of the (stochastic) Mixture Tuned Matched Filter (MTMF).

2.5.1 Finite Target Matched Filter (FTMF)

We have previously illustrated geometric and stochastic algorithms for solving both fully resolved and sub-pixel detection problems. The real problem of interest is one that involves sub-pixel targets where we used a standard linear mixing model approach (LMM) in conjunction with a orthogonal projection operator or estimate of the background covariance via stochastic means. In using the projection approach, we rely upon the existence of one or more linear projections of the spectral data in which the desired target and the suppressed clutter background are well separated from one another [43]. The more difficult task is to discern targets that are close to linear mixtures of background endmembers. Using an orthogonal projection method that suppresses the background (*e.g.*, matched filter) will more than likely suppress the target, if it is indeed an intimate mixture with the background. Furthmore, something like a matched filter will produce large outputs, not only for actual targets, but for any other spectral anomaly that has a significant projection (*e.g.*, a bright or saturated pixel). What is needed is a more stochastic approach that takes into account the spectral variability of the target imbedded in the background mixture. An approach that is more selective at rejecting false alarms than using a traditional matched filter and is based on the stochastic mixing model (SMM) (*cf.* Sec. 2.2.2), is called the finite target matched filter (FTMF) [44, 51].

This algorithm estimates the background mean and covariance from the data while assuming the target is deterministic with hypothetical variability (covariance) due to sensor noise. An observed sub-pixel target $\mathbf{x}$ (with an abundance a) is modeled as a stochastic mixture of the target and background with normal probability distribution according to

$$p(\mathbf{x}|a) \triangleq \mathcal{N}(\boldsymbol{\mu}(a), \boldsymbol{\Sigma}(a)) \tag{2.64}$$

where we are using the symbol $\triangleq$ to mean "defined as". The linear mixtures of means and covariances are then defined as

$$\boldsymbol{\mu}(a) = a\boldsymbol{\mu}_t + (1-a)\boldsymbol{\mu}_b \tag{2.65}$$

$$\boldsymbol{\Sigma}(a) = a^2\boldsymbol{\Sigma}_t + (1-a)^2\boldsymbol{\Sigma}_b \tag{2.66}$$

where $\boldsymbol{\mu}_t$ is the mean of the target and $\boldsymbol{\mu}_b$ is the mean of the background.

From this we establish the binary hypothesis

$$\begin{aligned} \mathcal{H}_0&: \quad a = 0, \text{ target absent} \\ \mathcal{H}_1&: \quad a > 0, \text{ target present} \end{aligned}$$

We can then establish a generalized likelihood ratio as

$$T_{FTMF}(\mathbf{x}) = \frac{\max\limits_{0\le a\le 1} p(\mathbf{x}|a)}{p(\mathbf{x}|a=0)} \tag{2.67}$$

where $\max_{0\le a\le 1}$ is the maximum likelihood estimate of the abundance a which is in the range $0 \le a \le 1$. The decision regions generated from Eq. (2.67) are not hyperquadric surfaces typically encountered in Gaussian hypothesis testing problems, but are instead truncated hypercone-like shapes that have a blunt "vertex" near

the pure target constituent and open out in the direction of the background [51]. The result is better selectivity, as compared to a traditional matched filter, due to a *nonlinear* decision boundary.

2.5.2 (Stochastic) Mixture Tuned Matched Filter (MTMF)

The SMM based FTMF is similar to the popular Mixture Tuned Matched Filter (MTMF) [5] found in the ENVI software environment [40]. This algorithm operates on MNF transformed data (*cf.* Appendix B) and uses a standard (stochastic) Matched Filter (MF) along with an added output infeasibility (INF) score. The score is based on feasibility mixtures between the composite background and the target spectrum. Likely target candidates have large matched filter values and small infeasibility scores. The ENVI software then displays a plot of SMF v.s. INF where the end user manually selects appropriate target pixels based on the criteria previously described.

As of this writing, there is no known published derivation of the MTMF first introduced by Boardman [5], other than an exchange of ideas through private conversation [7]. In this section we derive the MTMF based on the standard Spectral Matched Filter (SMF).

We start off by transforming the data into a space where the noise is isotropic with unit variance. The algorithm that does this is called the maximum noise fraction (MNF) which is very similar to principal component analysis (PCA). In PCA, we rely on maximizing the band-to-band *total image variance*. When using the MNF transform, we maximize the signal relative to the noise (*i.e.*, signal-to-noise ratio (SNR)). More information on the MNF transform can be found in Appendix B.

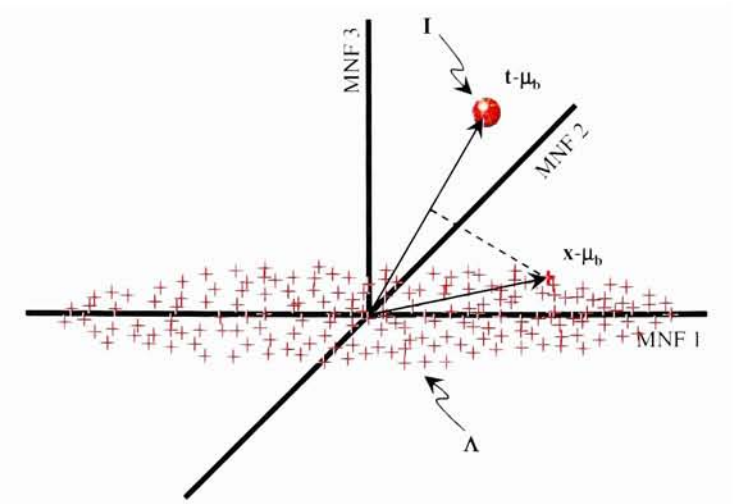


Figure 2.10: Illustration of spectral matched filter (SMF) parameters in 3 bands of an MNF space. The background data has covariance $\mathbf{\Lambda}$ while the target's covariance is the identity $\mathbf{I}$, since target variability was strictly based on noise in the original space. The target location relative to the background is exaggerated for illustrative purposes.

In this MNF space we can express the spectral matched filter as

$$T_{SMF}(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\mu}_b)^T \mathbf{\Lambda}^{-1} (\mathbf{t} - \boldsymbol{\mu}_b) \tag{2.68}$$

where $\boldsymbol{\mu}_b$ is the background mean, $\mathbf{x}$ is the pixel spectrum of interest, $\mathbf{\Lambda}$ is the spectral covariance matrix of the scene (background), $\mathbf{t}$ is the target and all the values are in MNF space where $\mathbf{\Lambda}$ is a diagonal matrix of eigenvalues (which is equal to the variance) in the MNF bands (*cf.* Figure 2.10).

Equation (2.68) is analogous to the previously derived matched filter of Eq. (2.22), except that now we are operating in a decorrelated space. This form of the equation is reminiscent of the *scalar product* found in linear algebra.

We can factor Eq. (2.68) using the square-root decomposition $\mathbf{\Lambda}^{-1} = \mathbf{\Lambda}^{-1/2}\mathbf{\Lambda}^{-1/2}$

which yields

$$T_{SMF}(\mathbf{x}) = \left(\mathbf{\Lambda}^{-1/2}(\mathbf{x} - \boldsymbol{\mu}_b)\right)^T \left(\mathbf{\Lambda}^{-1/2}(\mathbf{t} - \boldsymbol{\mu}_b)\right). \tag{2.69}$$

The SMF in MNF space gives us, in scalar form, how much of the test pixel is like the target through the projection of the whitened pixel, $\mathbf{x}$ onto the whitened target vector $\mathbf{t}$. A more convenient measure of this would be to normalize the scalar quantity by the whitened target vector squared. That is,

$$a = \frac{T_{SMF}(\mathbf{x})}{(\mathbf{t} - \boldsymbol{\mu}_b)^T \mathbf{\Lambda}^{-1} (\mathbf{t} - \boldsymbol{\mu}_b)} = \frac{\left(\mathbf{\Lambda}^{-1/2}(\mathbf{x} - \boldsymbol{\mu}_b)\right)^T \left(\mathbf{\Lambda}^{-1/2}(\mathbf{t} - \boldsymbol{\mu}_b)\right)}{(\mathbf{t} - \boldsymbol{\mu}_b)^T \mathbf{\Lambda}^{-1} (\mathbf{t} - \boldsymbol{\mu}_b)}. \tag{2.70}$$

This approach is similar to the *scalar projection* found in linear algebra except here we are not explicitly normalizing by the magnitude or Euclidian length. The reason for this is so that when the pixel of interest is the target, we get an abundance value of 1 (though a can take on values greater or less than one). This normalized quantity is analogous to the abundance derived for the matched filter in Eq. (2.23).

If we let $\alpha = (\mathbf{t} - \boldsymbol{\mu}_b)^T \mathbf{\Lambda}^{-1} (\mathbf{t} - \boldsymbol{\mu}_b)$ then we can write,

$$a = \frac{\left(\mathbf{\Lambda}^{-1/2}(\mathbf{x} - \boldsymbol{\mu}_b)\right)^T}{\alpha^{1/2}} \frac{\left(\mathbf{\Lambda}^{-1/2}(\mathbf{t} - \boldsymbol{\mu}_b)\right)}{\alpha^{1/2}}. \tag{2.71}$$

Upon analysis of Eq. (2.71), we can see that the data or test pixels(s), $\mathbf{x}$ and target $\mathbf{t}$ could also be viewed in a normalized *whitened* space, $\mathbf{w}$. The transformed image data, for example, can then be represented as

$$\mathbf{w} = \frac{\mathbf{\Lambda}^{-1/2}(\mathbf{x} - \boldsymbol{\mu}_b)}{\alpha^{1/2}}. \tag{2.72}$$

This normalized space is mean-centered, normalized by the expected standard deviation in each band (whitened) and further normalized by the square-root of the whitened target vector squared. Similarly, the target $\mathbf{t}$ can be represented as

$$\mathbf{w}_t = \frac{\mathbf{\Lambda}^{-1/2}(\mathbf{t} - \boldsymbol{\mu}_b)}{\alpha^{1/2}}. \tag{2.73}$$

Thus the abundance in the normalized whitened space can then be expressed as

$$a = \mathbf{w}^T \mathbf{w}_t. \tag{2.74}$$

As it turns out, the abundance, a is exactly the *scalar projection* of $\mathbf{w}$ onto $\mathbf{w}_t$. By definition, the scalar projection is defined as

$$a_{\text{proj}} = \frac{\mathbf{w}^T \mathbf{w}_t}{\|\mathbf{w}_t\|} \tag{2.75}$$

where $\|\mathbf{w}_t\|$ is the magnitude of $\mathbf{w}_t$. However, using the normalization α in our previous equations results in $\|\mathbf{w}_t\| = 1$. Therefore we have

$$a_{\text{proj}} = \frac{\mathbf{w}^T \mathbf{w}_t}{1} = a. \tag{2.76}$$

To calculate the abundance *vector* we simply multiply the scalar abundance by the unit vector, $\hat{\mathbf{w}}_t$. That is

$$\mathbf{a} = a\hat{\mathbf{w}}_t = \mathbf{w}^T \mathbf{w}_t \frac{\mathbf{w}_t}{\|\mathbf{w}\|} = (\mathbf{w}^T \mathbf{w}_t)\mathbf{w}_t \tag{2.77}$$

as can be seen in Figure 2.11.

Using an MNF transform yields a background covariance of $\mathbf{\Lambda}$ (*i.e.*, a diagonal matrix comprised of the eigenvalues (λ_j) from the noise-whitened background data). In going to $\mathbf{w}$-space we are scaling each band by

$$\alpha^{-1/2}\mathbf{\Lambda}^{-1/2}. \tag{2.78}$$

This produces a background covariance $\mathbf{S}_{wb}$ of

$$\mathbf{S}_{wb} = \left(\alpha^{-1/2}\mathbf{\Lambda}^{-1/2}\right)^T \mathbf{\Lambda} \left(\alpha^{-1/2}\mathbf{\Lambda}^{-1/2}\right) = \frac{\mathbf{I}}{(\mathbf{t} - \boldsymbol{\mu}_b)^T \mathbf{\Lambda}^{-1} (\mathbf{t} - \boldsymbol{\mu}_b)} = \frac{\mathbf{I}}{\alpha} \tag{2.79}$$

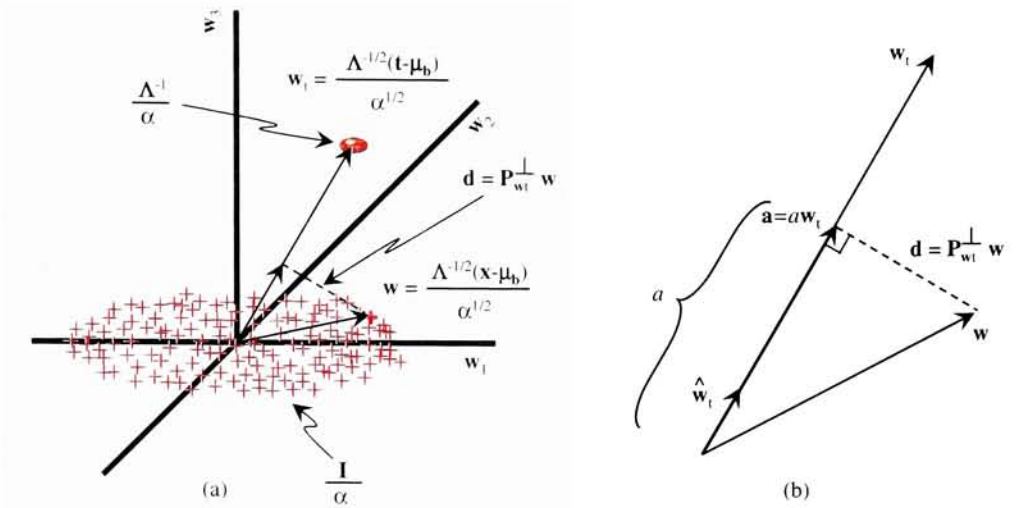


Figure 2.11: Illustration of SMF in normalized whitened **w**-space. (a) Shows the projection of a sample pixel **w** onto the target vector $\mathbf{w}_t$ to yield the abundance a. We also illustrate the orthogonal projection **d**. Additionally, the background data now has the covariance of a scaled identity matrix (circular distribution) while the target data covariance is a scaled version of the diagonal background data. (b) Illustrates the quantities associated with the abundance in more detail.

and target covariance $\mathbf{S}_{wt}$ of

$$\mathbf{S}_{wt} = \left(\alpha^{-1/2}\mathbf{\Lambda}^{-1/2}\right)^T \mathbf{I}\left(\alpha^{-1/2}\mathbf{\Lambda}^{-1/2}\right) = \frac{\mathbf{\Lambda}^{-1}}{(\mathbf{t}-\boldsymbol{\mu}_b)^T\mathbf{\Lambda}^{-1}(\mathbf{t}-\boldsymbol{\mu}_b)} = \frac{\mathbf{\Lambda}^{-1}}{\alpha} \tag{2.80}$$

where **I** is the variation in the target due to noise $\mathbf{\Sigma}_N$, which in an MNF space is the identity matrix (see Appendix B). These covariances are also illustrated in Figure 2.11.

If the abundance, a represents how "target-like" a vector is, then the projection onto the hyper-plane perpendicular to the target should represent how "un-target-like" or how much the test vector **x** is like the background. The operator that projects the target vector $\mathbf{w}_t$ onto the subspace orthogonal to the target is express

as

$$\mathbf{P}^{\perp}_{\mathbf{w}_t} = \mathbf{I} - \mathbf{w}_t\mathbf{w}_t^{\dagger} \tag{2.81}$$

where $\mathbf{w}_t^{\dagger} = (\mathbf{w}_t^T\mathbf{w}_t)^{-1}\mathbf{w}_t^T$which is the pseudo-inverse of $\mathbf{w}_t$. Operating on the i^{th} $\mathbf{w}$ vector with $\mathbf{P}^{\perp}_{\mathbf{w}_t}$ yields the vector $\mathbf{d}_i$ which describes the "un-target-like" component of $\mathbf{w}$. That is

$$\mathbf{d}_i = \mathbf{P}^{\perp}_{\mathbf{w}_t}\mathbf{w}_i \tag{2.82}$$

as shown in Figure 2.11. The magnitude or Euclidean length of $\mathbf{d}_i$ in this form can be expressed as

$$\|\mathbf{d}_i\| = \left[(\mathbf{P}^{\perp}_{\mathbf{w}_t}\mathbf{w}_i)^T(\mathbf{P}^{\perp}_{\mathbf{w}_t}\mathbf{w}_i)\right]^{1/2} = \left[\mathbf{w}_i^T\mathbf{P}^{\perp}_{\mathbf{w}_t}\mathbf{w}_i\right]^{1/2} = \left(\mathbf{w}_i^T\mathbf{d}_i\right)^{1/2} \tag{2.83}$$

because the projection matrix $\mathbf{P}^{\perp}_{\mathbf{w}_t}$ is both symmetric and idempotent (*i.e.*, $\mathbf{P}^{\perp}_{\mathbf{w}_t} = (\mathbf{P}^{\perp}_{\mathbf{w}_t})^T = (\mathbf{P}^{\perp}_{\mathbf{w}_t})^2$ (see Appendix D). We can also compute the magnitude of $\mathbf{d}$ by using either vector substraction or the Pythagorean theorem. Using vector subtraction $\mathbf{d}_i = \mathbf{w}_i - \mathbf{a}_i$ we have

$$\|\mathbf{d}_i\|_{\text{vs}} = (\mathbf{d}^T\mathbf{d})^{1/2} = \left[(\mathbf{w}-\mathbf{a})^T(\mathbf{w}-\mathbf{a})\right]^{1/2} = \left[\mathbf{w}^T\mathbf{w} - \mathbf{a}^T\mathbf{a}\right]^{1/2} \tag{2.84}$$

since $\mathbf{w}^T\mathbf{a} = \mathbf{a}^T\mathbf{w} = \mathbf{a}^T\mathbf{a}$. Lastly, we can compute $\mathbf{d}$ in terms of scalars using the Pythagorean theorem. This yields

$$\|\mathbf{d}_i\|_{\text{pythag}} = (\|\mathbf{w}\|^2 - \|\mathbf{a}\|^2)^{1/2} = \left[\mathbf{w}^T\mathbf{w} - \mathbf{a}^T\mathbf{a}\right]^{1/2}. \tag{2.85}$$

Thus far we have projected the test pixel onto the target and established an abundance term and a corresponding perpendicular difference term in a normalized whitened space. Based on these two parameters (and a threshold) we determine whether or not the background pixel of interest is a target or not. We can refine our decision metric further by analyzing the geometric shape formed between the target

and background distributions [*cf.* Figure 2.12(a)]. This shape can be described as a cone with its width at the base defined by the variability in the background data and its width at the apex defined by the variability in the target data. Geometric analysis tells us that the variability in **d** should be small when the target abundance is large and relatively larger when the target abundance is small, which reflects greater variability in the background.

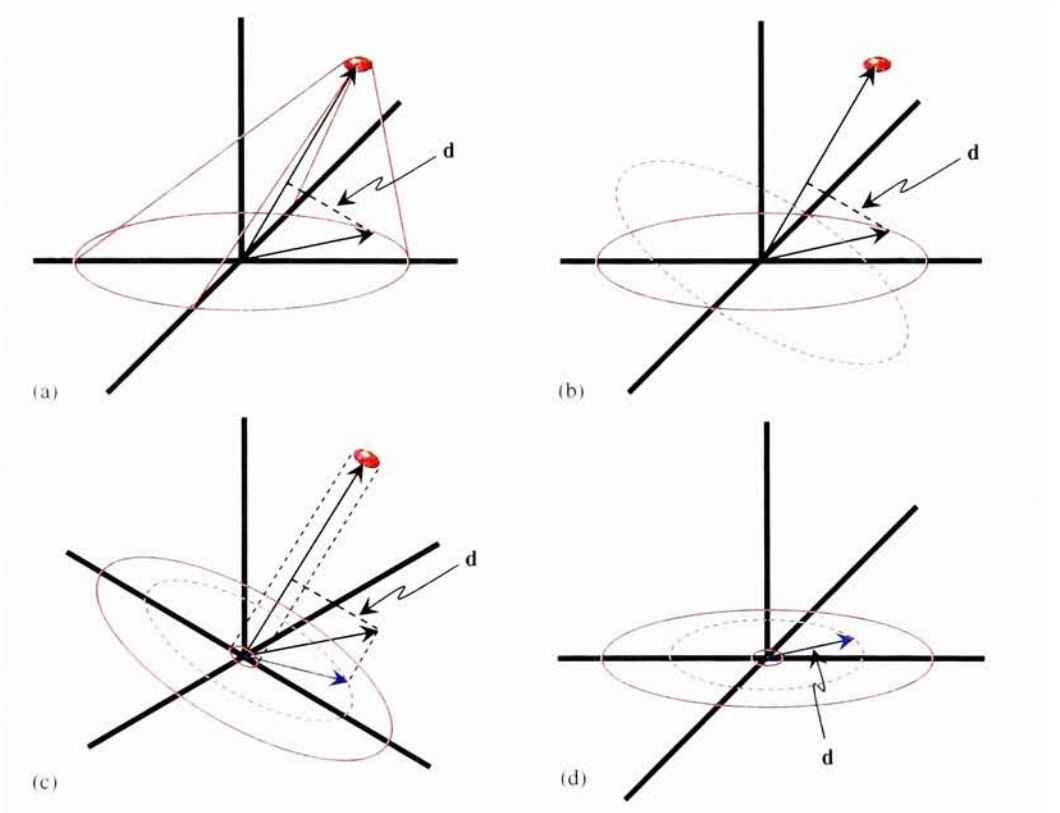


Figure 2.12: (a) Geometric cone formed by target and background data. (b) Rotation and (c) projection of the target and background data in (a). (d) The projected **d** vector.

We can use our projection operator to transform the data space so as to make

the target vector $\mathbf{w}_t$ perpendicular to the background data [*cf.* Figure 2.12(b)]. This means that when we operate on $\mathbf{w}_t$, we should get

$$\mathbf{P}^{\perp}_{\mathbf{w}_t}\mathbf{w}_t = (\mathbf{I} - \mathbf{w}_t\mathbf{w}_t^{\dagger})\mathbf{w}_t = \mathbf{0}. \tag{2.86}$$

When we transform the background covariance in whitened space we have

$$\mathbf{S}_{db} = (\mathbf{P}^{\perp}_{\mathbf{w}_t})^T\mathbf{S}_{wb}(\mathbf{P}^{\perp}_{\mathbf{w}_t}) = \frac{(\mathbf{P}^{\perp}_{\mathbf{w}_t})^T\mathbf{I}(\mathbf{P}^{\perp}_{\mathbf{w}_t})}{\alpha} = \frac{\mathbf{P}^{\perp}_{\mathbf{w}_t}}{\alpha}. \tag{2.87}$$

Similarly, we can transform the whitened target covariance as

$$\mathbf{S}_{dt} = (\mathbf{P}^{\perp}_{\mathbf{w}_t})^T\mathbf{S}_{wt}(\mathbf{P}^{\perp}_{\mathbf{w}_t}) = \frac{(\mathbf{P}^{\perp}_{\mathbf{w}_t})^T\mathbf{\Lambda}^{-1}(\mathbf{P}^{\perp}_{\mathbf{w}_t})}{\alpha} = \frac{\mathbf{P}^{\perp}_{\mathbf{w}_t}\mathbf{\Lambda}^{-1}\mathbf{P}^{\perp}_{\mathbf{w}_t}}{\alpha}. \tag{2.88}$$

In doing so, the $\mathbf{d}$ vector will be parallel to (and projected onto) the transformed background data [*cf.* Figure 2.12(c)]. The result is a set of *contour lines* that are representative of the magnitude of $\mathbf{d}$ [*cf.* Figure 2.12(d)]. Conceptually, our contour lines (prior to projection) can be thought of as forming the bounds of a cone outlining the magnitude of the vector $\mathbf{d}$. We can describe the variation about this cone based on a *linear mixture* of the (rotated) target and background standard deviations (or variances) modulated by the abundance [*cf.* Figure 2.13(a)]. This linear mixture can be described as

$$s = at + (1 - a)b. \tag{2.89}$$

If we wish to find the variance of this mixture we have

$$\begin{aligned} \sigma^2(s) &= \sigma^2(at + (1-a)b) \\ &= \sigma^2(at) + \sigma^2((1-a)b) - 2\sigma^2(at, (1-a)b) \\ &= a^2\sigma^2(t) + (1-a)^2\sigma^2(b) - 2a(1-a)\sigma^2(t,b) \end{aligned} \tag{2.90}$$

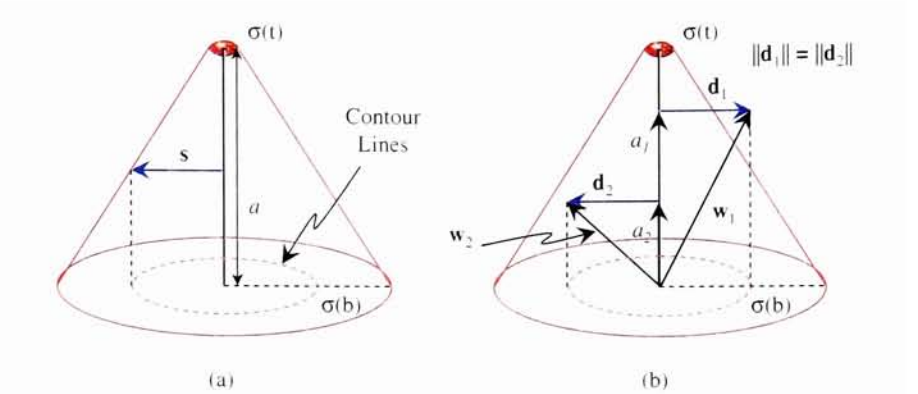


Figure 2.13: (a) Cone and contour lines defined by **s** which is a function of $a, \sigma(t)$ and $\sigma(b)$. (b) Illustration of two samples pixels, $\mathbf{w}_1$ and $\mathbf{w}_2$. with different abundances but the same magnitudes for the **d** vectors. However, the infeasibility will be greater for pixel 1 than pixel 2.

where a complete derivation can be found in Appendix C. If we have independent random variables, then the covariance between t and b is zero. Therefore we have

$$\sigma^2(s) = a^2\sigma^2(t) + (1-a)^2\sigma^2(b). \tag{2.91}$$

We can then ratio the magnitude of the vector **d** with $\sigma^2(s)$ to obtain a scalar metric $\hat{d}$, proportional to how many scaled standard deviations a pixel is from the expected target/bakground mixture. That is

$$\hat{d} = \frac{||\mathbf{d}||}{\sigma^2(s)}. \tag{2.92}$$

If we band-normalize $\hat{d}$, then we have what is referred to as the infeasibility (INF) which is described as

$$INF_{\text{single band}} = \frac{\hat{d}}{p} \tag{2.93}$$

where p is the number of bands included in the MNF space. Figure 2.13(b) shows the infeasibility concept as hyper-cones about the target vectors. The diagonals of $\mathbf{S}_{dt}$ and $\mathbf{S}_{db}$ yield the variance values, $\sigma^2(t)$ and $\sigma^2(b)$, needed in Eq. (2.91).

To include the interactions of all the bands in Eq. (2.91), we can re-define $\sigma^2(s)$ as

$$\mathbf{\Sigma}_s = a^2\,\mathbf{S}_{dt} + (1-a)^2\,\mathbf{S}_{db}. \tag{2.94}$$

We can then proceed with the infeasibility metric as

$$INF = \frac{\mathbf{d}^T\mathbf{\Sigma}_s^{-1}\mathbf{d}}{p}. \tag{2.95}$$

The MTMF algorithm has been applied to the sub-pixel target detection problem [5] which showed that it could be used to effectively reduce the number of false positives by rejecting pixels with high infeasibility values.

It should be pointed out that this is a decision space with 2 inputs, the results from the spectral matched filter and the infeasibility metric. Therefore, an actual two dimensional decision criteria based on these parameters would need to be developed. Lastly, the derivation presented is based on personal conversation [7] only and has yet to be implemented. Future efforts will check the validity of such a derivation while simultaneously comparing it to the version found in the ENVI [40] software environment.

2.6 Background Summary

In this section we introduced the concept of detection from a theoretical point of view followed by a discussion of methods (structured and unstructured) used to estimate variability in image backgrounds. We then discussed some of the more common algorithms used for both full and sub pixel target detection. The section on the development and usage of *target spaces* leads us in a new target detection

direction that incorporates physics modeling. This was the foundation of the illumination invariant method. Lastly we discussed algorithms that were different from traditional detectors in that they either combined geometrical and statistical information about the imagery or simply used an approach entirely different from traditional detection techniques.

This research sees potential in exploring the novel method of detection that incorporates physics based modeling. In doing so, we must additionally develop detectors or detection schemes that can adapt to such an approach. A modeling and detection scheme that is based on fundamental physics may prove to be advantageous and useful to the hyperspectral detection community.

"The important thing in science is not so much to obtain new facts as to discover new ways of thinking about them."

William Bragg

"Your theory is crazy, but it's not crazy enough to be true."

Niels Bohr

3

New Approach

Previous studies [14] using the Physics Based Modeling (PBM) approach generated target spaces with thousands of spectra in them. These methods did not take advantage of some of the physical process and interactions involved when generating the space. In Sec. 3.1 we illustrate methods on how to "refine" this space by employing the use of algorithms and some common *a priori* knowledge to help predict MODTRAN input parameters, given a scene of interest. We then use this target space in a newly developed detection algorithm called the Physics Based Structured InFeasibility Target-detector (PB-SIFT) which incorporates a Physics Based Orthogonal Subspace Projector (PBosp) and a Structured Infeasibilty Projector (SIP), both of which are outlined in Sec. 3.2.

3.1 Improved Modeling of a Target Space

In Sec. 2.4.1 we described a method, based on the physics model of Eq. (2.49), that generated a sensor-reaching radiance target *space*, with variation due to physical effects. Previous studies [14] have generated such target spaces with over 28,000 spectra in them, some 17,000 of which were physically realizable. These target spaces were extensive because of the large number of MODTRAN input parameters varied, particularly the gas concentrations. Atmospheric gas parameters such as $O_3, CO, CO_2, CH_4, N_2O, O_2$, were varied along with solar zenith angle, aerosol type, sensor altitude, and water vapor.

Although this method encompasses many of the parameters, the combinatorial variations between the gasses may not be necessary. This is because some of the concentrations of these "ideal" gasses should co-vary with each other. By representing the molecular gas species as a single modification in the generation of a look-up-table (LUT), a series of MODTRAN runs that represent improbable conditions can be eliminated from the simulation runs. The following sections describe such an idea by narrowing the number of input parameters to a handful that can adequately characterize the atmosphere.

3.1.1 Variation in Visibility (VIS)

Healey and Slater [14] varied 4 aerosol models. That is, rural, urban, maritime, and desert extinction. The major difference in these models is the particle size distribution. For most studies, however, one can select an appropriate aerosol model based on a given data set. A rural extinction model represents aerosols mostly related to areas not strongly affected by urban or industrial sources. An urban

extinction is a mixture of 80% rural aerosol with 20% soot-like aerosols and is most appropriate for high density urban industrial areas. Maritime extinction is mainly used for cases over water or continents under a prevailing wind from the ocean. It consists of two components, one from sea spray and another from rural continental aerosol.

Once the aerosol model is selected, we further vary the aerosol *number density* (concentration of number of particles) using the (horizontal) *visibility* [km] parameter in MODTRAN. The visibility (VIS) scales the number density of particles of a certain type that exist in the atmosphere. Since this parameter directly impacts atmospheric scattering, it will have a major effect on sensor reaching spectra in the VNIR region (400-1000 nm). It should be noted, however, that at high visibilities (greater than 30 km) the overall impact (or variation) is reduced.

Since visibility is dependant on weather conditions, a thick haze might be representative of a 15 km visibility. Similarly, a moderate haze might be a 20 to 30 km visibility while a clear atmosphere might have a visibility of 40 to 100 km. A more accurate initial estimate of visibility can be obtained from area surface observations. These can be obtained from local airports or the National Oceanic and Atmospheric Administration (NOAA) where records of perceived horizontal visibility and general surface weather conditions (including variations) are kept. Additionally, an estimate of visibility can be obtained using an atmospheric compensation algorithm called FLAASH (described below). Here, the algorithm generates a scene-average visibility (*i.e.*, aerosol/haze amount) using a dark pixel reflectance ratio method [21, 22].

3.1.2 Variation in Ground Topography

The problem with varying each individual constituent gas parameter (*i.e.*, O_3, CO, CO_2), for example, is that we are not taking advantage of actual physical interactions among molecules. Ultimately, by varying the gas parameters, we are after the amount of *molecular absorption*. Using the "combinations" method will certainly generate physically un-realizable spectra, in addition to unnecessary runs. Since a multitude of dependencies exist within the atmospheric physical process, a more elegant solution would be to simply vary the *path length* from the sensor to the ground. This variation would control how much molecular absorption there is over the path length. In general, the sensors altitude (as measured from the ground to the platform) is fairly constant and usually known. This altitude is set to the MODTRAN parameter "H1". We then look down (MODTRAN ANGLE=180) in the direction of the ground to altitude location "H2". It is this H2 parameter we vary, which can be though of as a variation in ground topography. This is explained in more detail in Sec. 4.1.3.

This noted, there are some gases, on larger scales, that very seasonally. One gas in particular is carbon dioxide. The mixing ratios from 4 different observatories can be seen in Figure 3.1. Here we see an overall mixing ratio increase from the early 70's to present day. Additionally, we can see small oscillations that are synonymous with season fluctuations.

An estimate of sensor elevation (as measured relative to sea level) or altitude can be obtained by knowing the sensor platform height. Ground topography variation and estimates might be obtain from elevation information about the region of interest. The significance of this parameter becomes increasingly important as the topography of the ground becomes more mountainous or undulating. Conversely,

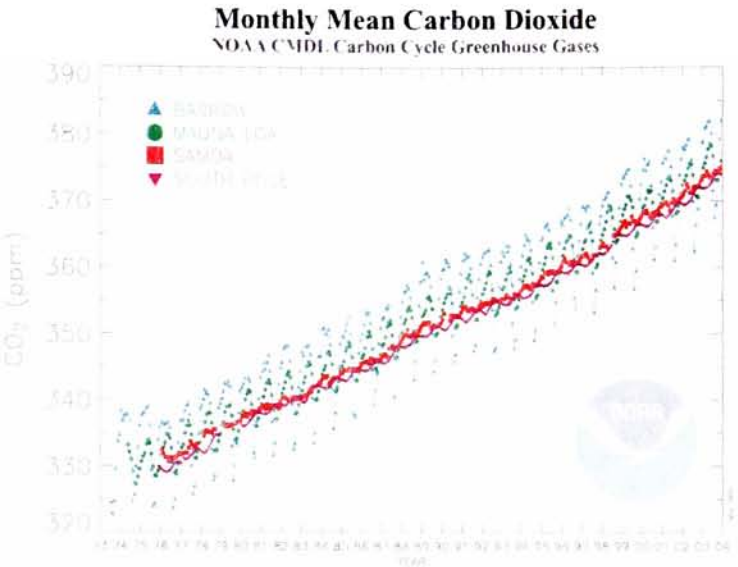


Figure 3.1: Atmospheric carbon dioxide mixing ratios determined from the continuous monitoring programs at the 4 NOAA CMDL baseline observatories [18].

the importance is diminished as the terrain becomes more flat.

3.1.3 Water Vapor (WV)

Water vapor is by far the largest contributor to absorption in the atmosphere, as can be seen in Figure 3.2. This constituent directly impacts atmospheric transmission, in the form of attenuation of the upwelling radiance, and ultimately affects sensor-reaching radiance. Water vapor is usually measured as relative humidity (RH), as a function of altitude, and then converted to a measure of precipitable water in g/cm^2 or *centimeters* of water in a given column. Ideally, one would like to have an atmospheric water vapor profile above the target of interest. This can be achieved by launching a radiosonde balloon at the site. This information can then be used as

input to MODTRAN so as to generate sensor reaching radiance spectra that reflect the conditions at the time of image acquisition.

In general, it would behoove us to understand water vapor since it has such a significant impact on sensor-reaching radiance. In the event that radiosonde data is not available, we need to formulate a best guess as to what the atmospheric water content is. If given a specific hyperspectral data set, we can take some of the guess work out of the process by using *physically derived* values of integrated column water vapor obtained through use of estimation algorithms [26, 45].

Water Vapor Estimation Algorithms

In cases where measurements of WV are not available or there is a need to validate the applicability of a given integrated WV measurement to a site of interest, we can take advantage of in-scene techniques that yield image derived estimates of total water column vapor. By total we mean the amount of water vapor in the atmosphere vertically integrated.

The Continuum Interpolated Band Removal (CIBR) [26] algorithm is one such program that can be used to estimate water vapor. It is based on looking at the 940 nm absorption feature and relating its depth to water vapor on a per pixel basis. To obtain variability here, one would simply look at the variability in the calculated water vapor map.

An improvement to the CIBR model is the Atmospheric Pre-corrected Differential Absorption (APDA) [45] algorithm. It combines a partial atmospheric correction with a differential absorption technique. It works by iteratively correcting for the atmospheric path radiance term leading to the retrieval of water vapor. Again, this process yields per pixel values where variability can be interpreted from the water

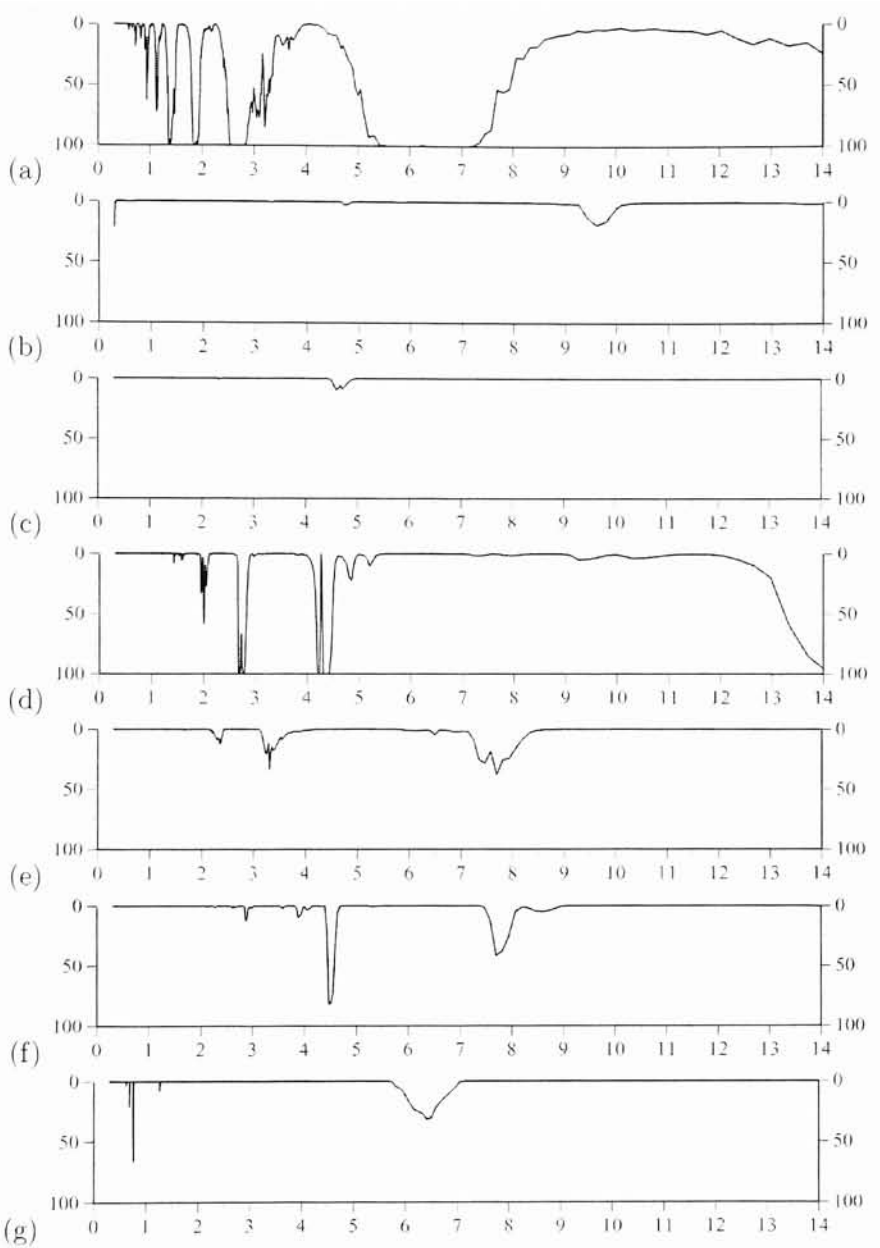


Figure 3.2: Absorption spectra of (a) H_2O, (b) O_3, (c) CO, (d) CO_2, (e) CH_4, (f) N_2O, and (g) O_2 as a function of wavelength $[\mu m]$. We can see that of all the atmospheric constituents, H_2O is the largest contributor to absorption.

vapor map. Studies comparing these two methods [15] have shown that they both did a reasonable job of estimating water vapor column abundance, which allowed atmospheric compensation to be applied.

Finally, one could use a water vapor map generated as a by-product of a model-based atmospheric compensation algorithm such as the Atmospheric REMoval (ATREM) [10] program or the Fast Line-of-sight Atmospheric Analysis of Spectral Hypercubes (FLAASH) [1] algorithm.

ATREM is software developed by the University of Colorado for retrieving scaled surface reflectance from hyperspectral data using a radiative transfer model. A useful by-product from the ATREM software is an image of the column water vapor which was removed from the hyperspectral data. FLAASH is similar in that it also is a physics based atmospheric compensation algorithm. Specifically, it is a MODTRAN-4 based atmospheric compensation software package developed by the Air Force Research Laboratory (AFRL) and Spectral Sciences, Inc. (SSI). It provides accurate, physics based derivations of apparent surface reflectance through derivation of atmospheric properties such as surface albedo, surface altitude, water vapor column, aerosol and cloud optical depths, as well as surface and atmospheric temperatures from hyperspectral data. The water vapor estimates are calculated on a per pixel basis using radiance averages from two sets of channels. One set of channels is an absorption set centered at a water band, and the other is a reference set taken from the edge of the water band.

Water Vapor and MODTRAN

In MODTRAN, the input parameter that effects the amount of water absorption is the *water vapor column scaling factor*. For example, a scalar value of 2.0 simply

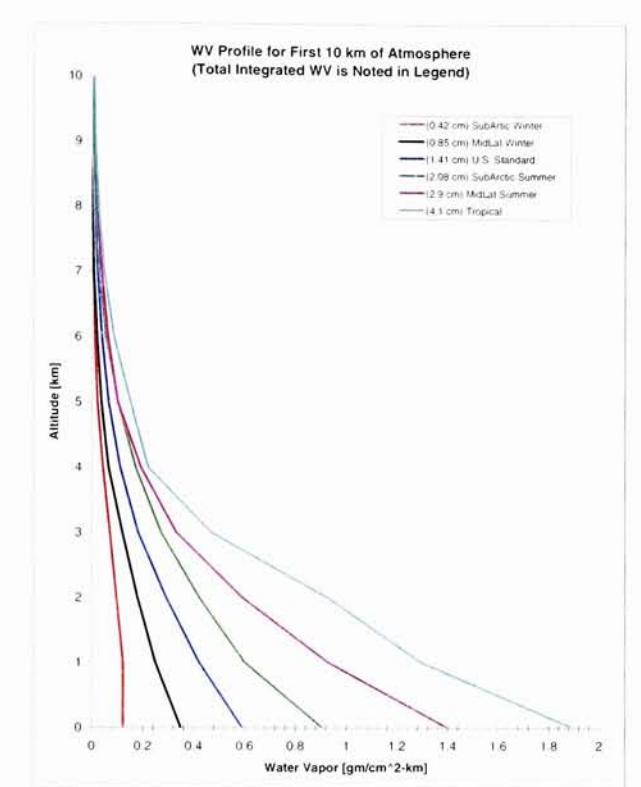


Figure 3.3: MODTRAN atmospheric water vapor profiles for the first 10 km of the atmosphere.

doubles the default water vapor column, which is set by the atmospheric model. MODTRAN uses one of six default water column profiles, depending on which atmospheric model the user selects. The first 10 km of these profiles can be seen in Figure 3.3 along with (vertically) integrated column amounts.

The water vapor amounts are usually reported in units of $[atm-cm]$ or $[g/cm^2]$. The conversion, using dimensional analysis, from one to the other is expressed as

$$1\,atm-cm\left(\frac{1\,mole}{22413.83\,atm-cm^3}\right)\left(\frac{18.015\,g}{1\,mole}\right)=\left[8.037E^{-4}\frac{g}{cm^2}\right] \qquad (3.1)$$

where the first term is molar volume of water at standard temperature and pressure (STP, 1 mole, 1 atm) and the second term is the molar mass for 1 molecule of water. The units of $[g/cm^2]$ can be also be thought of as an *interaction depth* or a measure of the path length that helps in the understanding of absorption. For example, at any point along a path, the number of interactions is proportional to the density times the path length. If one were to add up all of these interactions along the particle's path, we would get a number that is proportional to the total absorption. Since density (ρ) has units of $[g/cm^3]$ and the path length (l) is in units of $[cm]$, then the interaction depth (ID) will have units of $[g/cm^2]$. That is

$$ID = \rho\, l \quad \left[\frac{g}{cm^2}\right]. \tag{3.2}$$

Often we see the units $[g/cm^2]$ simply expressed as $[cm]$. Especially when stating total integrated column water vapor values. This re-expression is made possible because of the density definition of water at STP. Knowing the interaction depth and density of water $[1\, g/cm^3]$, we can calculate the path length as

$$l = \frac{ID\,[g/cm^2]}{[1\, g/cm^3]} = ID\,[cm] \tag{3.3}$$

which for water, is simply the interaction depth in units of $[cm]$. For example, as divers know, a depth of 10 meters in water provides an additional atmosphere (1 atm) of pressure. In other words, 10 meters of water will provide the same absorption as the entire thickness of the atmosphere.

If we know the water profile being used, then we also know the total integrated column amount for that profile. This total amount would correspond to a water vapor scale factor of one in MODTRAN. To obtain spectra that correspond to other integrated water amounts, MODTRAN simply linearly scales the profile total.

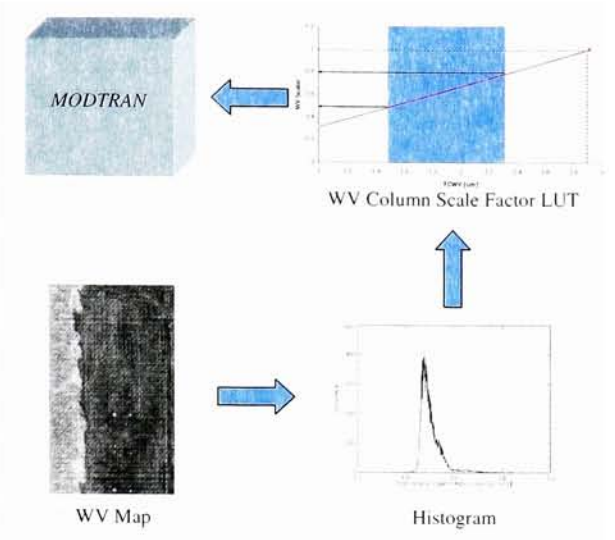


Figure 3.4: Process of obtain MODTRAN water vapor scale factors using results from physics-based water vapor estimation routines.

Therefore, the process would involve obtaining a water vapor map from an estimation algorithm, determining the appropriate range of water vapor values (*e.g.*, using a histogram), and then convert the range of values to a range of water vapor column scale factors to use as input to MODTRAN. This process is illustrated in Figure 3.4.

3.1.4 Variation in Target Orientation

Past applications of the invariant method varied up to 8 unique *zenith* angles (not to be confused with target orientation) [14, 50]. In our approach the zenith angle is known (and fixed) because we assume the user has some *a priori* knowledge about when the imagery was collected (*i.e.*, time of day and location). Even if the spectral character of the target, as seen by the sensor through the atmosphere,

is modeled correctly in MODTRAN, we still may have issues related to spectral magnitude. This can manifest itself as a time of day variation, target-to-sensor angle or orientation variation, or both. In this research we factor in an additional parameter that allows the user to vary the *orientation* of the target relative to the Sun.

This variation in target orientation, or more precisely illumination, is simply a modulation of the direct solar term to account for projected area effects (*cf.* Figure 3.5). We implement this as a target rotation angle relative to the zenith angle determined in MODTRAN, for the time of day. If the rotation is zero, then there is no adjustment of the direct term. However, a positive or negative rotation angle implies that we have rotated the target toward or away from the Sun, respectively. The relation for this modulation of the direct term is

$$E_{s\_new}(\lambda) = \frac{E_s(\lambda)}{\cos\sigma'} \cos\sigma_{new} \tag{3.4}$$

where $E_s(\lambda) = E'_s(\lambda)\cos\sigma'$, $\sigma_{new} = \sigma' - \sigma_{rot}$ and σ_{rot} is the user specified angle of rotation. The zenith angle σ' comes from the MODTRAN output. It is important to note here that $E_s(\lambda)$ in this notation comes directly from MODTRAN and already has the zenith angle attenuation applied to it. We differentiate this value from $E'_s(\lambda)$ which is the direct solar term with out any angular effects.

As an example, if the zenith angle was 60 degrees and we specify a rotation angle of $\sigma_{rot} = -10$ degrees, then we have

$$E_{s\_new}(\lambda) = \frac{E_s(\lambda)}{\cos 60} \cos(60 - (-10)) = 0.68 E_s(\lambda) \tag{3.5}$$

which says we have effectively reduced the illumination onto the target by 32%.

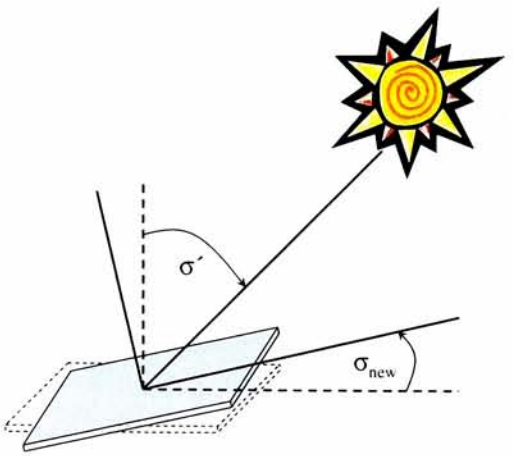


Figure 3.5: Illustration showing the difference between the zenith angle σ' and target orientation angle σ_{new}.

3.1.5 Variation in Downwelled Radiance

In addition to direct solar illumination onto the targets surface we also have solar photons that originate from scattering in the atmosphere. We refer to this downward scattering as *downwelled radiance*, or, in the visible, as *skylight*. More often than not, the target is *not* "completely" exposed to the entire hemisphere above, which is sky. In reality, if the target has any slope or if there are adjacent objects (*e.g.*, a tree or building) obstructing the sky dome, the downwelled radiance onto the target will be reduced, as can be seen in Figure 3.6. In general, the altering of the downwelled term will have a larger impact in the visible than at longer wavelengths, due to the nature of atmospheric scattering.

The original implementation of the invariant method used a physical sensor reaching radiance equation that did not directly allow for modulation of the downwelled term. Instead they used a binary constant (K) that accounted for occluding bodies in the solar to surface path. Reduced downwelled effects occurred simply

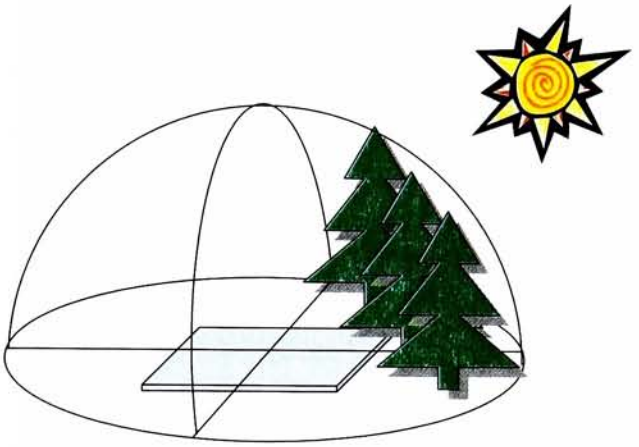


Figure 3.6: Illustration of shape factor concept where target is not completely exposed to the hemisphere above due to an obscuration such as a tree, for example.

because the zenith angle (which was varied) was large. A more accurate approach would be to directly modulate the downwelled term by a scale factor ranging from 0 to 1. This scale factor is often referred to as the shape factor, F [46]. A shape factor of one implies the target is completely exposed to the sky hemisphere. Conversely, a shape factor of zero means the target is completely occluded from the sky dome. Values between zero and one imply partial occlusion (*e.g.*, a tree canopy).

3.1.6 Summary of Target Space Modeling

In the previous sections, we set forth on differentiating our method from the original implementation of the invariant method. This new approach does not require the user to vary as many atmospheric parameters. We have refined this process to include variation from a handful of parameters such as visibility (VIS), ground topography, and water vapor scalar (WV). The trade off is that the user is required to know some *a priori* information about the scene. That is, the type of aerosol

model to use (urban, rural, etc.), the sensor altitude, day of year, latitude, longitude, and time of day. Most of these *a priori* parameters are easy to obtain and are usually collected at the time of image acquisition. Lastly, we have introduced two new terms that were not included in the original invariant algorithm. These include target orientation or projected area effects and shape factor, which is used to control the amount of downwelled radiance. These last two parameters can be seen in our new sensor reaching radiance equation which is of the form

$$L_s(\lambda) = \left[\frac{E_{s\_new}(\lambda)}{\pi}\tau_1(\lambda) + FL_d(\lambda)\right]\tau_2(\lambda)r(\lambda) + L_u(\lambda) \tag{3.6}$$

where $E_{s\_new}(\lambda) = \frac{E_s(\lambda)}{\cos\sigma'}\cos\sigma_{new}$ as described in Eq. (3.4) and r is the measured reflectance of the target.

3.2 PB-SIFT Algorithm

In Sec. 2.5.2 we analyzed and derived the stochastic MTMF. This algorithm is based on the statistics of the image (covariance) and requires a pre-processed MNF image, which additionally requires an estimate of the image noise. In this research, we develop a geometric equivalent that does not require image noise estimates. Our algorithm captures data variability through use of endmembers or basis vectors (*cf.* Sec. 2.2.3). The developed algorithm is called the Physics Based Structured Infeasibility Target-detector (PB-SIFT) which incorporates a Physics Based OSP (PBosp) detector and a measure of how un-target-like a pixel is, called the Structured Infeasibility Projector (SIP).

3.2.1 Physics Based OSP Matched Filter (PBosp)

If the target and backgrounds are described using geometric techniques then the application of a detector based on vector geometry is most appropriate. We first describe the data simplex with a set of endmembers or basis vectors while using the pixel model defined in Eq. (2.27). These endmembers are then used as column vectors to populate a background descriptor matrix, $\mathbf{B}$. We can use this information about the background space to develop a least squares (orthogonal) projection operator that tells us how unlike the background a pixel appears. This non-background-like behavior $\mathbf{e}$ can be described as

$$\mathbf{e} = \mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x} \tag{3.7}$$

where the orthogonal projection operator $\mathbf{P}_{\mathbf{B}}^{\perp} = \mathbf{I} - \mathbf{B}\mathbf{B}^{\dagger}$ and $\mathbf{B}^{\dagger}$ is the pseudo-inverse (in a least squares sense) of $\mathbf{B}$ defined as $\mathbf{B}^{\dagger} = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T$. The quantity $\mathbf{B}\mathbf{B}^{\dagger}$ is known in matrix theory as the *projection matrix* and has the properties of being idempotent and symmetric. We can then take the non-background-like weighting vector $\mathbf{e}$ and project it onto a target spectrum $\mathbf{t}$. This produces the detector

$$T(\mathbf{x}) = \mathbf{t}^T\mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x} \tag{3.8}$$

which is a type of geometric matched filter that is the un-normalized Orthogonal Subspace Projection (OSP) detector [13]. Instead of estimating the background covariance, we assume the linear mixing model (LMM) $\mathbf{x} = \mathbf{B}\mathbf{b}$ and describe the background space by populating the matrix $\mathbf{B}$ with endmembers and estimating their fractions $\mathbf{b}$, by projecting the pseudo-inverse of $\mathbf{B}$ onto the pixel $\mathbf{x}$. The orthogonal operator $\mathbf{P}_{\mathbf{B}}^{\perp}$ acting on $\mathbf{x}$ not only does this "projection" but gives us the error $\mathbf{e}$ associated with modeling $\mathbf{x}$ as a background pixel (*i.e*, a linear mixture of backgrounds) where our estimate of $\mathbf{x}$ is termed $\hat{\mathbf{x}}$ (*cf.* Figure 3.7). That is,

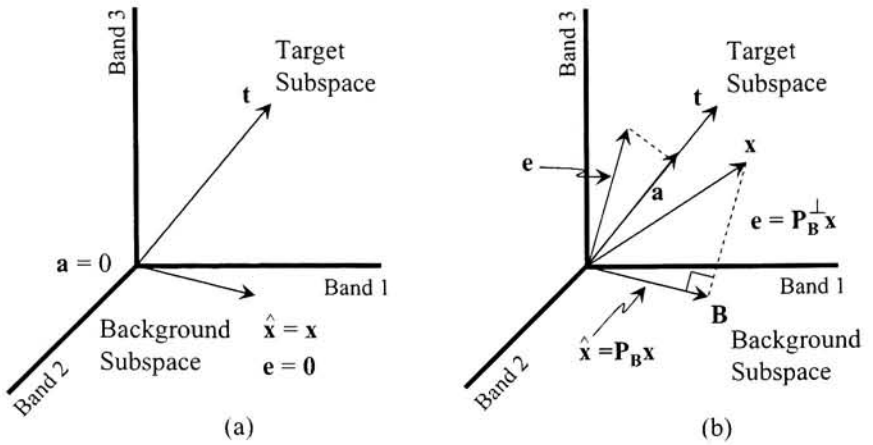


Figure 3.7: Illustration of background mixing model as applied to test pixel $\mathbf{x}$. (a) The test pixel $\mathbf{x}$ produces an error vector $\mathbf{e}$ of zero because the background model provides a good fit. Thus the pixel is very background-like making the abundance $\mathbf{a}$ zero as well. (b) The background model is less appropriate for test pixel $\mathbf{x}$. The large error implies the test pixel is not background-like which can produce a larger projections onto the target vector producing significant abundances.

$$\begin{aligned} \mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x} &= (\mathbf{I} - \mathbf{B}\mathbf{B}^{\dagger})\mathbf{x} \\ &= \mathbf{x} - \mathbf{B}\underbrace{(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{x}}_{\hat{\mathbf{b}}} \\ &= \mathbf{x} - \mathbf{B}\hat{\mathbf{b}} \\ &= \mathbf{x} - \hat{\mathbf{x}} = \mathbf{e} \end{aligned} \tag{3.9}$$

We are essentially testing the appropriateness of the background model on pixel $\mathbf{x}$. If the error is small then the test pixel is background-like because the model provides a good fit. Thus the pixel is more background and less target which makes the abundance small as well [*cf.* Figure 3.7(a)]. On the other hand, if the error (specifically the orthogonal projection) is large then the background model is less

appropriate and we say that the test pixel is not background-like and may look like target [*cf.* Figure 3.7(b)]. Actual implementation usually requires a threshold or cut off value for the test statistic in order to judge whether image pixels are target or not.

We wish to further generalize Eq. (3.8) so as to incorporate more than a single target vector. The motivation for this is the fact that the product of physics based modeling produces not one, but hundreds of target-like vectors forming a *target space*. We can ultimately reduce this space to a handful of basis vectors but we still need a detector that can address multiple instances of a target vector.

This is done by simply projecting our $\mathbf{e}$ vector onto the target *space*, $\mathbf{T}$ (where $\mathbf{T}$ is a matrix of target vectors) rather than a single vector, $\mathbf{t}$. We use the projection operator $\mathbf{P_T}$ such that our detector now takes on the form

$$T_{\text{Gosp}}(\mathbf{x}) = \|\mathbf{P_T}\mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x}\| \tag{3.10}$$

where we call T_{Gosp} the Generalized OSP detector because we are generalizing to *more than one* target vector. Additionally, $\mathbf{P_T} = \mathbf{T}\mathbf{T}^{\dagger}$ where $\mathbf{T}^{\dagger}$ is the pseudo-inverse of $\mathbf{T}$ defined as $\mathbf{T}^{\dagger} = (\mathbf{T}^T\mathbf{T})^{-1}\mathbf{T}^T$ (Note: this inverse can also be computed using an SVD). If we let $\mathbf{e} = \mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x}$ then we can write Eq. (3.10) as

$$T_{\text{Gosp}}(\mathbf{x}) = (\mathbf{e}^T\mathbf{P_T}\mathbf{e})^{1/2}. \tag{3.11}$$

If the projection operator is composed of a *single* vector (*i.e.*, $\mathbf{P_T} \Rightarrow \mathbf{P_t} = \mathbf{t}\mathbf{t}^{\dagger}$), then we can relate it to the un-normalized OSP detector as

$$T_{\text{Gosp\_single}}(\mathbf{x}) = \|\mathbf{P_t}\mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x}\| = \frac{\mathbf{t}^T\mathbf{P}_{\mathbf{B}}^{\perp}\mathbf{x}}{\|\mathbf{t}\|}. \tag{3.12}$$

In this research, we implement a normalization of Eq. (3.10) based on the *average*

target-space spectrum $\mathbf{t}_{avg}$. That is

$$T_{\text{PBosp}}(\mathbf{x}) = \frac{\|\mathbf{P_T P_B^{\perp} x}\|}{\|\mathbf{P_T P_B^{\perp} t}_{avg}\|} \tag{3.13}$$

which produces positive values. The lower range of this detector is zero, which implies zero target abundance in the test pixel. However, the upper range can take on values larger than one since we are normalizing by the average target space vector. In the event that $\mathbf{x} = \mathbf{t}_{avg}$, we have $T_{\text{PBosp}}(\mathbf{x}) = 1$.

3.2.2 Structured Infeasibility Projector (SIP)

The detector of Eq. (3.8) is a type of matched filter where we are weighting a target spectrum with an (orthogonal) projection of the test pixel with the background space. However, large abundances can also be generated by any spectral pixel that has a significant projection (*e.g.*, a bright or saturated pixel) thus producing false positives. This type of false positive we call *leakage*. Figure 3.8 illustrates this concept where we have two different pixels producing the same abundance *vector* $\mathbf{a}$ which is defined as

$$\mathbf{a} = a\hat{\mathbf{t}} = a\frac{\mathbf{t}}{||\mathbf{t}||} \tag{3.14}$$

where $\hat{\mathbf{t}}$ is a unit vector and $a = \mathbf{t}^T\mathbf{P_B^{\perp} x}$ from Eq. (3.8).

For this reason, we would like to use an additional *infeasibility* metric that helps us detect the presence of false positives relative to information about the target and background spaces.

In Eq. (3.13) we established a measure of how target-like a pixel was, based on a geometric background description. We now wish to obtain a measure $\mathbf{d}$ of how un-target-like $\mathbf{x}$ is using an orthogonal projection onto the subspace perpendicular

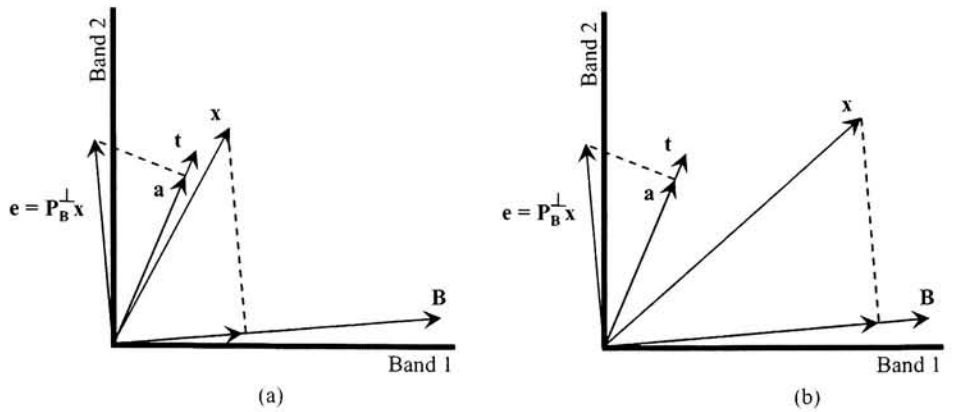


Figure 3.8: Illustration of a possible false positive using a matched filter. In (a) the image pixel $\mathbf{x}$ is close to the target and results in an abundance vector $\mathbf{a}$. In (b), however, it is questionable as to whether the image pixel is close to the target even though it produces the same abundance-like value as in (a).

to the target vector (*cf.* Figure 3.9). That is the target infeasibility is expressed as

$$T_{SIP}(\mathbf{x}) = ||\mathbf{d}|| = ||\mathbf{P}_{\mathbf{T}}^{\perp}\mathbf{x}|| \tag{3.15}$$

where $\mathbf{P}_{\mathbf{T}}^{\perp} = \mathbf{I} - \mathbf{T}\mathbf{T}^{\dagger}$ and $\mathbf{T}^{\dagger}$ is the pseudo-inverse of $\mathbf{T}$ as previously defined. This magnitude is synonymous with Euclidian distance. The target signature can come from a spectral library (assuming reflectance data) or be image derived. In this research vectors of $\mathbf{T}$ originate from a (radiance) target space created using physics based modeling (PBM). This approach is similar to the one derived in Eq. (2.82) except here we are working with multiple target vectors from a target space.

We now have a vector $\mathbf{d}$ that can be used to aid in the identification of pixels that have similar orthogonal projections onto the background but different projections onto the subspace orthogonal to the target space, as can be seen in Figure 3.9(b). Upon closer analysis of Figure 3.9(b) we notice that there are potentially a *family* of pixels (denoted by the dotted line connection the tip of $\mathbf{e}$ and $\mathbf{x}$) that can produce

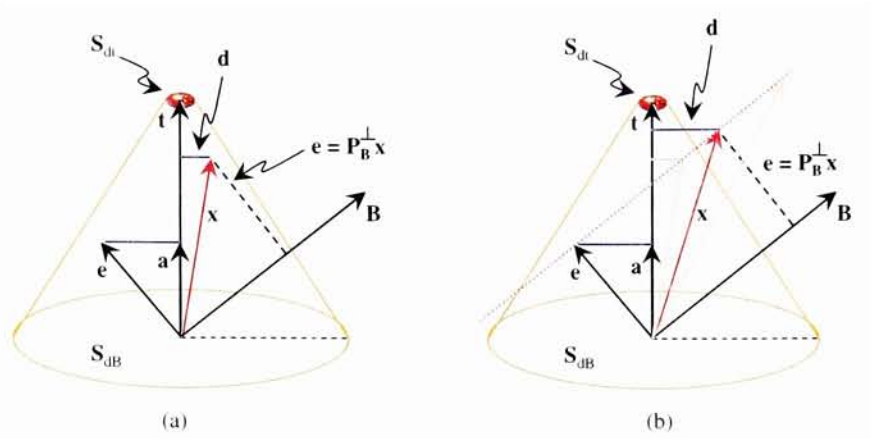


Figure 3.9: Illustration of the structured infeasibility concept as hyper-cones. For simplicity here, we assume the target space is a single vector (*i.e.*, $\mathbf{T} \Rightarrow \mathbf{t}$). (a) The projection of a pixel $\mathbf{x}$ onto the background $\mathbf{B}$ produces vector $\mathbf{e}$ which is then projected onto target $\mathbf{t}$ to produce an abundance-like vector $\mathbf{a}$. Additionally, we can generate a vector projection orthogonal to the target termed $\mathbf{d}$. (b) Shows that we can get the same abundance vector for a "family" of possible pixels, all of which, however, have different $\mathbf{d}$ values, which we can take advantage of in generating a structured infeasibility projector (SIP).

the same orthogonal projection onto the background $\mathbf{B}$, but with different $\mathbf{d}$ values.

3.2.3 2-Dimensional Decision Space

The use of Eq. (3.13) and (3.15) produces a two dimensional decision space that exploits the behavior of an image pixel relative to its target-like and un-target-like behavior. In this space we seek out pixels that have a large abundance and small infeasibility [*cf.* Figure 3.10(a)]. This is exactly the type of decision map the mixture tuned matched filter produces [40] in which, at this point, user intervention would be required to select the most probably target pixels.

In our 2 dimensional decision space we additionally recognize the anticipated behavior of potential target pixels. We realize that the *mean* orthogonal projection of

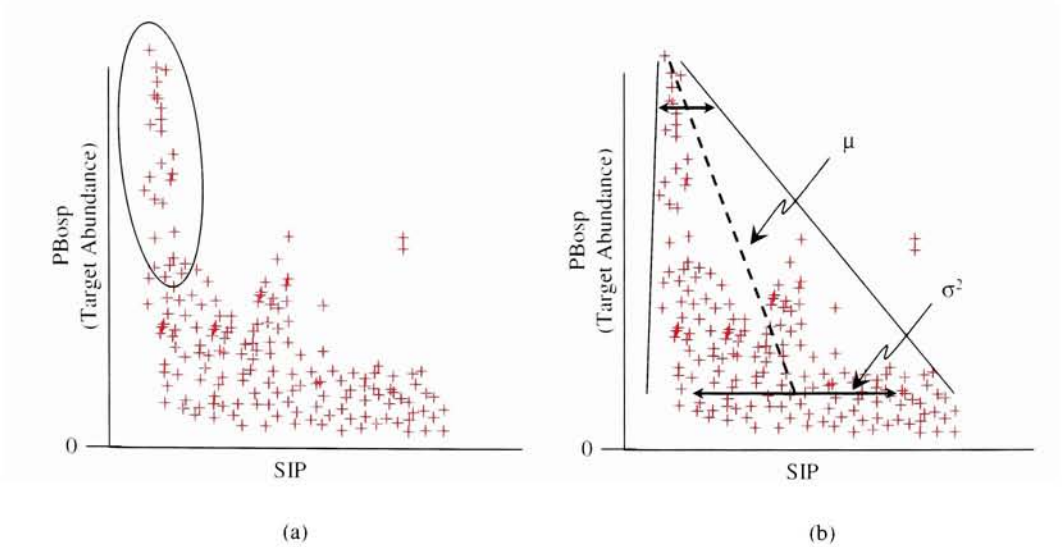


Figure 3.10: Illustration of 2 dimensional decision map. In (a) the most probable target pixels are those with large abundances and small infeasibilities (SIP values). In (b) we show the expected statistical behavior of the SIP values as a function of target abundance.

the test pixel onto the target space (the mean SIP value) will get closer to zero as the pixel becomes target-like (*i.e.*, larger abundance). This can be seen in Figure 3.10(b) where, as abundance increases, the mean orthogonal projection vector gets smaller in length. This is due to the variability of the data as a function of abundance. We additionally realize that, in general, the background space will have larger variability than the target space. Because target pixels are associated with large abundances, the variance forms a skewed cone around the abundance driven mean [*cf.* Figure 3.10(b)].

Because of the skewness, a linear decision boundary along the SIP axis is impossible to define. This skewness can be removed by normalizing by the abundance

driven mean. That is

$$T_{SIP\_norm}(\mathbf{a}) = \frac{||\mathbf{d}||}{||\mathbf{a}||\mu_t + (1 - ||\mathbf{a}||)\mu_B} \tag{3.16}$$

where $\mathbf{a}$ is our abundance defined in Eq. (3.13), μ_t is the mean of the magnitudes of the orthogonally projected target vectors onto the target space and μ_B is the mean of the magnitudes of the orthogonally projected image pixels onto the target space. One of the problems with this approach is that the range of the abundance is not 0 to 1. This is because we are normalizing by the *average* target space vector. Another potential problem involves the mean of the target space. If the target space mean is not synonymous with the target pixel(s) mean, the skewness will not be removed. Overall, the decision boundary should take into account the abundance driven variability of the SIP values.

3.3 Hyperspectral Image Data

3.3.1 HYDICE FR I Data

The hyperspectral imagery that will be used through out this research is the HYDICE Forest Radiance I (FR I) data set which contains 210 spectral bands. This data set consists of a series of flight lines, over many days, where man made panels, vehicles, and objects were deployed along the ground and in an adjacent forest. The collection consisted of 3 phases or experiments. In phase I all target panels and objects were place in an "open" field where they were fully exposed. In phase II the same panels and objects were moved to an adjacent tree line where they were partially concealed due to tree line shadowing. Finally, in phase III all targets and objects were moved into the adjacent forest so as to be fully concealed or at least

Figure 3.11: 150 × 250 subset from the larger FR I, phase I, fully exposed data set.

hidden from direct sunlight.

For this research, imagery from the phase I and phase III experiments was used. Furthermore, only panels that were either fully exposed or fully concealed were considered in the detection study. Specifically, the flight lines used were termed run05 (fully exposed) and run25 (fully concealed). For the exposed imagery a 150 × 250 pixel image subsection was used (*cf.* Figure 3.11). A slightly larger 250 × 300 pixel subsection was extracted for testing from the fully concealed imagery (*cf.* Figure 3.12). These image sizes were selected because of run time issues with the current hardware used in this research. Anything larger produced run times that were not conducive to algorithmic testing and evaluation. A summary of the FR I collection parameters for both the fully exposed and concealed flight lines can be seen in Table 3.1.

3.3.2 FR1 Data Preprocessing

The units associated with the HYDICE FR I data are $[75 \cdot Wm^{-2}sr^{-1}\mu m^{-1}]$. In order to make the data compatible with the generated target spaces, everything

Figure 3.12: 250 × 300 subset from the larger FR I, phase III, fully concealed data set.

was converted to micro-flicks (μF). That is, a conversion factor of (100.0/75.0) was applied to the data to obtain units of $[\mu W cm^{-2} sr^{-1} \mu m^{-1}]$ or $[\mu F]$.

Additionally, not all the spectral bands are used during processing. Bad bands, due to low signal-to-noise (bands 1-5 and 200-210) and water absorption (bands 104-111 and 139-154) were removed. These removed bands can be seen in Figure 3.13. This left a total of 170 working spectral bands.

3.3.3 FR I Truth Masks and Target of Interest

In order to evaluate the performance of the detection algorithms, pixel truth masks of the exact locations of each target panel need to be made. The truth masks for the FR I exposed imagery (run05) were created by a group at MIT's Lincoln Laboratory and were contained in a processed version of the data known as the Canonical Data Set. This mask includes the pixel locations of all targets (full and sub pixel) for the "run05" flight line, as can be seen in Figure 3.14. The truth

Table 3.1: Summary of the FR I collection parameters for both the fully exposed and fully concealed flight lines.

	Exposed Imagery	Concealed Imagery
Date	8-24-1995	8-29-1995
Time Of Day	9:10 am EDT	2:37 pm EDT
Latitude	39.33469 N	~39.33 N
Longitude	76.27811 W	~76.27 W
Sensor Altitude	5146 ft (1.56 km)	~5000 ft
Scene Elevation	50 ft (0.015 km)†	50 ft (0.015 km)†

†Another obtained value for the scene elevation was 20 ft (0.006 km).

mask for the fully concealed image, however, was not available at the time of this research. Documentation from the ground truthing report did provide a rough schematic outlining the relative locations of various target panels. This was used as an aid in developing a *concealed* truth mask for the handful of targets used in this research. This labeled mask can be seen in Figure 3.15.

The target sought after in both the exposed and fully concealed flight lines was a green cotton fabric labeled F3. This panel represented 30 full and 5 sub pixels in the exposed image and approximately 18 pixels in the fully concealed image (12 pixels for F3a and 6 pixels for F3b). For this research, we attempt to identify all 35 exposed F3 pixels and all 18 fully concealed F3 pixels. It was found, however, that many other targets had a very similar reflectance to that of F3. Most similar was a green cotton target called F5 (*cf.* Figure 3.16). Still other materials made of a green fabric (T1 and T2) had similar spectral characteristics to that of F3, though F5 remained most similar.

For the qualitative analysis of the exposed and concealed imagery in Sec. 4.3.1

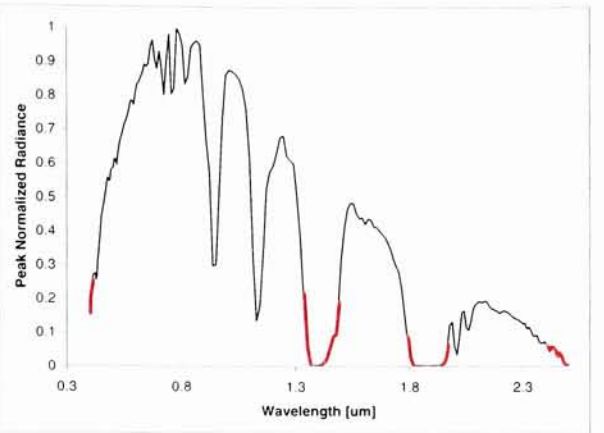


Figure 3.13: Peak normalized radiance spectrum showing regions (in red) where data was omitted leaving 170 working bands.

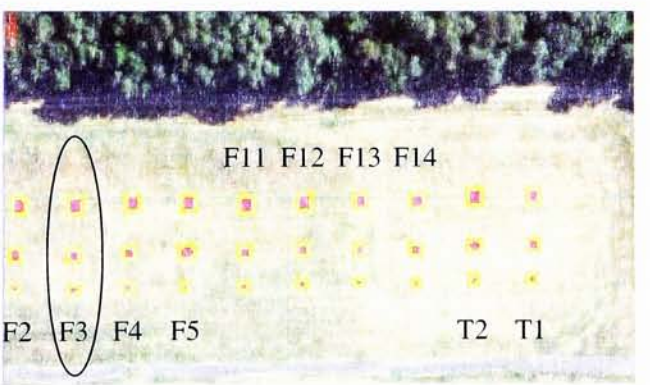


Figure 3.14: Sub section (150×250) of HYDICE FR I exposed image showing locations and labeling of target panels. The magenta pixels are associated with full pixels while the red pixels denote sub pixel targets. The yellow pixels are referred to as "guard" pixels and are not considered target pixels.

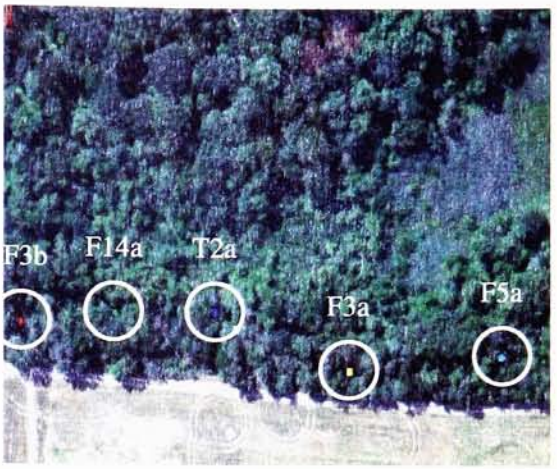


Figure 3.15: Sub section (250 × 3000) of HYDICE FR I fully concealed image showing locations and labeling of target panels. For this research we attempt to find the F3 panel which is denoted by the red and yellow pixels.

and 4.3.2, we show test statistic values for *all* pixels. However, for generation of ROC curves, we mask out target panel test statistic values associated with all but the F3 man-made target. We are interested in the algorithms ability to differentiate the F3 signature from other naturally occurring signatures. Later analysis, that generates 2 dimensional decision spaces, uses truth masks that algorithms false alarm on all objects.

3.4 Algorithm Comparison and Methods of Assessment

In order to quantify the newly developed detector we must compare its performance to algorithms that are currently used in the community. For this study, we refrain from using exotic detection algorithms in favor of those that are more "off the

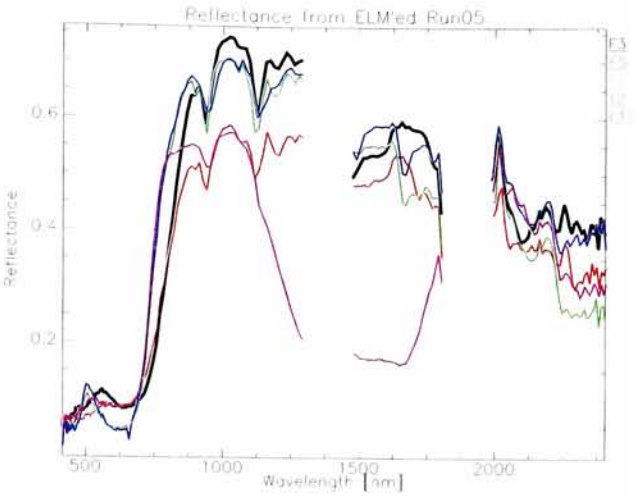


Figure 3.16: Plot showing ELM'ed derived reflectances for various targets that look similar to that of panel F3. It can be seen that panel F5 is most similar to F3, though slightly reduced in magnitude.

shelf'. This should, at the very least, establish a base line for our new algorithms performance. The algorithms to be compared include the Spectral Angle Mapper (SAM), Spectral Matched Filter (SMF), and the Mixture Tuned Matched Filter (MTMF), which is the SMF with an additional infeasibility score. These algorithms will be compared to our new Physics Based Structured InFeasibility Target-detector (PB-SIFT), which includes a Physics Based OSP (PBosp) detector and Structured Infeasibility Projector (SIP).

Algorithm assessment for this research will be performed in both a qualitative and quantitative manner. Qualitative methods are in the form of black and white test statistic images. For the PB-SIFT algorithm, analysis is made, in addition to the previous method, through use of a 2-dimensional scatter plot where various target pixels are labeled through use of truth masks.

Finally, quantitative analysis is performed using the familiar Receiver Operating Characteristic (ROC) curve, which pairs the detection rate against the false alarm rate. These curves are made possible because of available truth masks.

"No amount of experimentation can ever prove me right; a single experiment can prove me wrong."

Albert Einstein

4 Results

In the following chapter we report the results of applying our new methodology to the generation of target spaces as well as results from applying our new detector to hyperspectral data. Section 4.1 gives an overview of the target space generation process as well as how the various input parameters are used. This is followed by an explanation of how the target and background space endmembers are derived (*cf.* Sec. 4.2). In Section 4.3 we show detection results using the PBosp, SMF, and SAM detectors followed by Sec. 4.4 which shows detection results using the added SIP metric. This result is then compared the MTMF algorithm. Finally, Sec. 4.5 looks at the trends exhibited in 2 dimensional decision spaces as well as how saturated or bright pixels manifest themselves in these spaces.

4.1 Target Space Generation

In the following sections we introduce the core of this approach to target detection. That is, the generation of target spaces using physics based modeling. We start off by discussing how one can utilize MODTRAN, along with some external utilities and scripts, to generate hundreds of spectra. Unlike previous approaches using this technique, we require the user to know some common *a priori* knowledge about the scene. This information is discussed along with possible choices the user has.

The user also has the task of selecting the most appropriate values for parameters such as visibility, scene topography, and water vapor scale factor, all of which are varied and scene dependent. We provide explanation and guidance on how to more appropriately select these values even when information about the scene maybe unknown.

After the MODTRAN runs have been completed, we explain the process of assembling a look-up-table and factoring in additional terms such as target orientation, which effectively takes into account projected area effects, and shape factor, which helps to modulate the downwelled radiance term. These look-up-tables are then constructed for both our test image data sets.

Finally, an evaluation of such a target space is performed using a traditional linear mixing model approach. We compare a known image target pixel to that of linear combinations of target space vectors. The final RMS error assessment is made after converting this difference to an equivalent reflectance.

4.1.1 Creating Target Spectra Using MODTRAN

In this section we briefly state the process of setting up the MODTRAN runs along with some (user supplied) required information to complete the MODTRAN card deck.

The approach taken in creating hundreds of MODTRAN runs is one that generates a directory hierarchy based on the parameters varied: visibility, scene topography (though it is called "elevation", which we will explain below), and water vapor scalar (*cf.* Sec. 3.1). That is, a C-shell script is used to create a directory path appropriate for a given vis/ele/wv combination. For example, if we wanted to create a MODTRAN run for vis (or aerosol) = 20km, ele = 0.015km and wv = 0.5 we would (automatically) create a path that looked like `AEROSOL/20.0/ELEVATION/0.015/WATER/0.5/ALBEDO/0.0`. We use an `ALBEDO` of 0.0 in order to obtain the path radiance term, L_u otherwise the surface albedo is set to 1.0. Because of this variation in albedo, we essentially double the number of MODTRAN runs. This is explained in more detail in Sec. 4.1.4.

Once the directory structure is in place, all the MODTRAN runs are submitted to a workload management system for compute-intensive jobs called Condor [19]. Condor provides a job queueing mechanism, scheduling policy, priority scheme, resource monitoring, and resource management which makes the hundreds of MODTRAN runs, and this approach to target detection, possible. Upon completion of the submitted jobs, the tape6 and tape7 files are parsed so as to obtain the various terms in the sensor-reaching radiance equation [*cf.* Eq. (3.6)]. These spectral quantities are then resampled using a user supplied sensor response file. Finally, the resampled quantities are assembled into an ASCII look-up-table (LUT) to be use in post processing (*cf.* Sec. 4.1.4).

Required Information for MODTRAN Runs

In order to generate successful MODTRAN runs, based on the criteria set forth in Sec. 3.1, we need to know some information about the scene and sensor response ahead of time. Some of these scene dependant parameters are fixed and some are varied. The required parameters, along with some known values, are summarized in Table 4.1.

Table 4.1: Summary of exposed and fully concealed scene parameters, along with known values, required for a MODTRAN run.

	Exposed Image	Concealed Image
Atmospheric Model	Midlatitude Summer†	Midlatitude Summer†
Aerosol Model	Rural Extinction†	Rural Extinction†
Sensor Altitude	5146 ft (1.56 km)	$\sim$ 5000 ft
Day of the Year	236	241
Latitude	39.33469 N	~39.33 N
Longitude	76.27811 W	~76.27 W
Time of Day	9:10 am EDT	2:37 pm EDT
Visibility	Varied	Varied
Scene Topography or Elevation	Varied	Varied
WV Scale Factor	Varied	Varied

†An estimated parameter based on scene conditions and location.

4.1.2 Non-Varied MODTRAN Parameters

Of the many MODTRAN parameters that are used to create spectral output, we focus our attention on 10 scene dependant quantities (*cf.* Table 4.1). In this section

we discuss the 7 quantities that are not varied during the MODTRAN runs. These quantities, however, do require the user to estimate their values. We additionally discuss why the values were selected, the level of difficulty in knowing the value, and the impact of its uncertainty on detection

Atmospheric Model

In the event that one does not have radiosonde data, an atmospheric model must be selected. The atmospheric models alter atmospheric profiles for temperature ($°C$), pressure (mb), relative humidity (%), and molecular gases such as H_2O, O_3, CH_4, N_2O, and CO. For this research the atmospheric model used for the FR I exposed and fully concealed collects was *mid-latitude summer*. This was chosen based on the day of year (summer) and geographic location (mid-latitude) of the collection site. In general, this parameter is relatively easy to estimate if information about where and when the imagery was collected is known. At this time, however, the impact on detection, when the user selects an inappropriate atmospheric model, is unclear. This will be the topic of future work.

Aerosol Model

The type of aerosol model used for the MODTRAN runs was that of *rural extinction*. The main difference between all the aerosol models is the particle size distributions (*cf.* Sec. 3.1.1 for additional information). Based on these, MODTRAN computes the scattering properties of the atmosphere which drives the absorption due to scattering. The models essentially drive the spectral shape of the scattering (and the resulting absorption). The urban aerosols have particles sizes corresponding to urban soot from factories and vehicles, etc. while the rural aerosols have sizes related

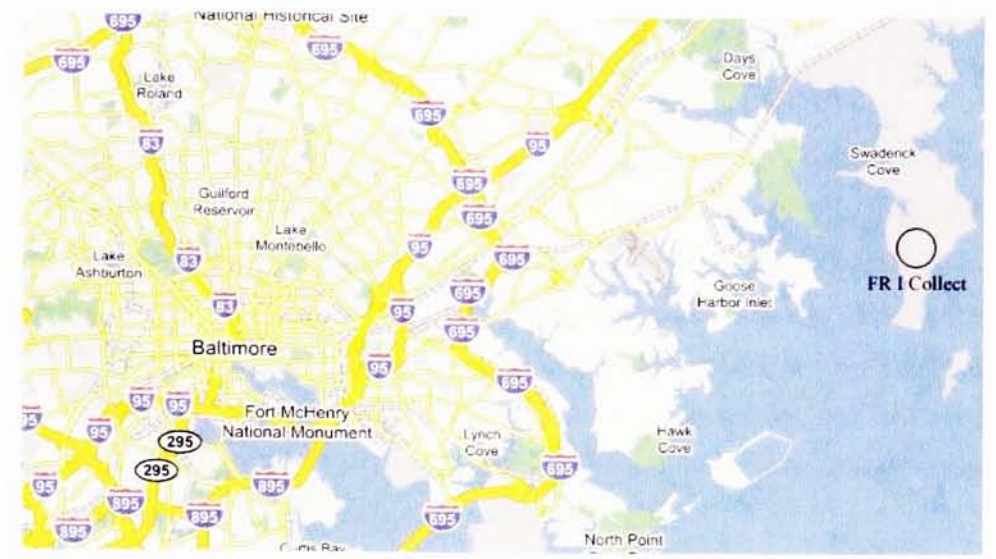


Figure 4.1: General location of collection site (middle right of image) relative to Baltimore, MD.

to dust-type particles. The maritime aerosols particles are based on particles from sea spray.

An extinction synonymous with an *rural* environment was chosen due to the location of the collection site (see Figure 4.1). From this, we can see that the collection area was directly east of Baltimore, MD. At at distance of 20 miles from Baltimore, the collection site should not be strongly influenced by industrial aerosol sources.

In general, the aerosol parameter can be easy to identify in the event that the collection is over water (maritime extinction), in the country (rural extinction) or over a large city (urban extinction). The impact it can have on detection results is lessened as the atmospheric visibility increases to values over 30 km (*cf.* Sec. 4.1.3).

Sensor Altitude

As defined in this research, sensor altitude is measured relative to the ground where as sensor elevation is with respect to mean sea level. The sensor altitude, as reported in Tables 3.1 and 4.1, was approximately the same for both the exposed and concealed collections. This information was obtained from flight logs generated during the collection campaign. More often than not, information about flying height is generally recorded and available. From this we choose not to explicitly vary this parameter. Additionally, variation in path length is more likely due to changes in terrain height rather than variation in sensor altitude.

Day of Year (DOY), Latitude, Longitude, Time of Day (TOD)

The day of year parameter, together with latitude, longitude, and time of day, help determine the position of the Sun in the sky. For this research all the before mentioned parameters were known at the time of the collection (*cf.* Tables 3.1). These parameters are usually recorded during collection time and are easy to obtain. In the event that this information is not completely known, one can obtain estimates by looking at shadow lengths and directions in the imagery to possibly obtain time of day information. Day of year and latitude/longitude information is so generally available that lack of knowledge of these parameters is not considered an issue.

4.1.3 Varied MODTRAN Parameters

In this section we continue our discussion on the remaining 3 scene dependant parameters, visibility, scene topography, and water vapor scale factor. As per Sec. 4.1.2 we address what values were used, why they were selected, the level of difficulty in knowing the values, as well as the impact of its uncertainty on detection. Addi-

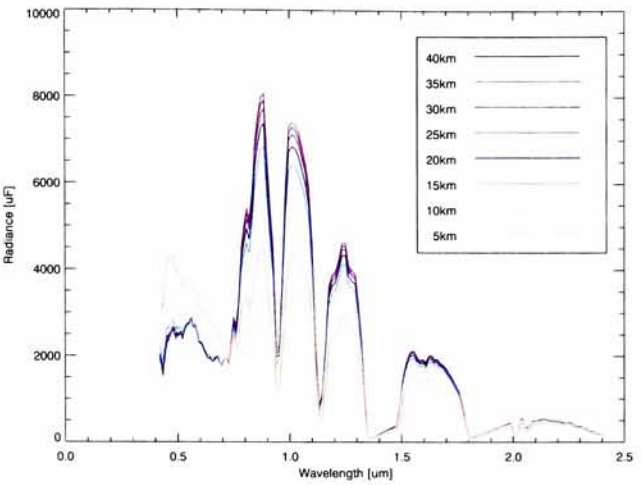


Figure 4.2: Variation in visibility, as seen by the sensor, using a rural extinction model. The radiance spectra include the target (F3) reflectance and are for a fixed elevation, water vapor scale factor, and TOD factors for the fully exposed collect. We can see that scattering dominates below 0.7 μm while absorption dominates at wavelengths above 0.7 μm.

tionally, these parameters vary and therefore have ranges associated with them. We further discuss the selection of such ranges in addition to sampling intervals.

Visibility

Visibility has one of the largest impacts on sensor reaching radiance because it is directly related to scattering (as discussed in Sec. 3.1.1). This makes it a key parameter to vary. We can see this variation and its impact on the sensor reaching model of Eq. (3.6) in Figures (4.2 – 4.4). The parameters and values used to generate these plots are those found in Table 4.1 for the fully exposed imagery. In addition, the sensor altitude was set to 1.56 km with a water vapor scale factor of 0.5.

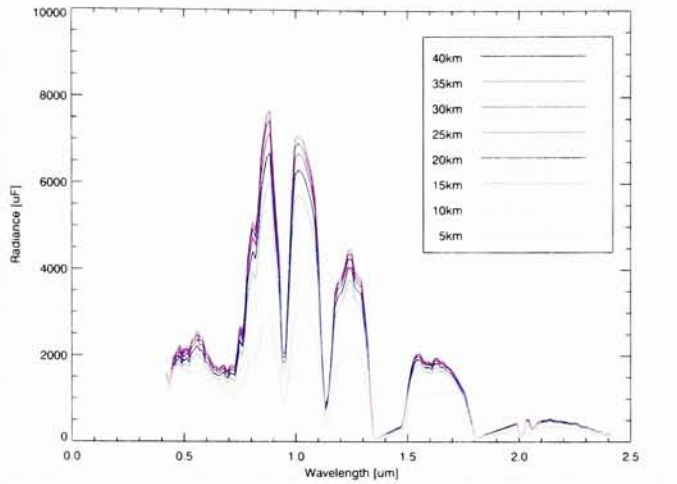


Figure 4.3: Variation in visibility, as seen by the sensor, using a urban extinction model. The radiance spectra include the target (F3) reflectance and are for a fixed elevation and water vapor scale factor, and TOD factors for the fully exposed collect.

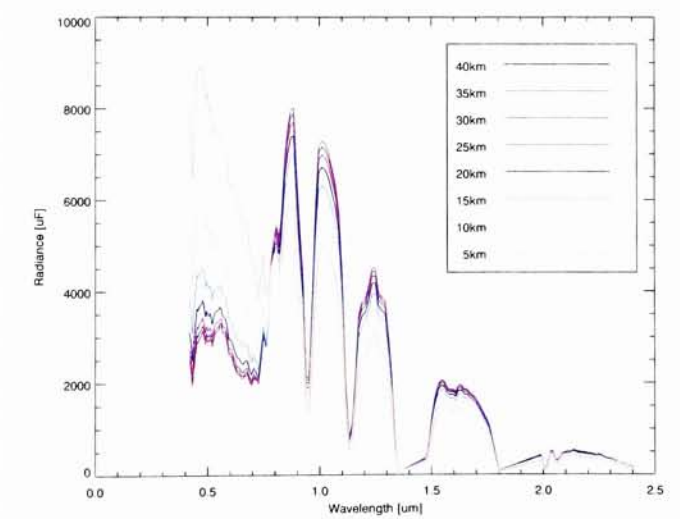


Figure 4.4: Variation in visibility, as seen by the sensor, using a maritime extinction model. The radiance spectra include the target (F3) reflectance and are for a fixed elevation and water vapor scale factor, and TOD factors for the fully exposed collect. We see the same trends as with the rural extinction model except for much more scattering.

Upon looking at Figure 4.2 using the rural extinction model, we see rather large radiances in the visible (*i.e.*, more scattering) and lower radiances in the NIR for the lower visibilities. In general, the lower visibilities (for all aerosol models) means lower atmospheric transmission and lower signal to the detector. However, lower visibility also means an overall increase in scattering in the atmosphere. We see that this effect is more pronounced at the lower blue wavelengths. Here we see that the amount of flux to the sensor from the VIS region is larger than that from the NIR. However, this increased flux is not from the target but rather the atmosphere due to increased scattering. Conversely, we see that as visibility increases we get an overall increase in atmospheric transmission and less scattering.

The trend using the urban extinction model, as seen in Figure 4.3, is slightly different. Here we see a *stronger* absorption component (compared to rural) associated with more absorbers due to soot and carbonized particles. This produces a lower and smaller range of radiance values in the visible (ignoring the 5km and 10km cases) but more variance when compared to the rural case.

When the maritime extinction model is used (*cf.* Figure 4.4) we can see a significant amount of scattering in the visible. This scattering is mainly due to water particles from sea spray based on a different particle size distribution than the rural and urban aerosol types.

For the FR I imagery, the perceived horizontal visibility as measured at a station in Baltimore, MD, ranged from 12 to 20 km for both the exposed and fully concealed collections. These estimates were obtained from surface weather observations obtained from the National Oceanic and Atmospheric Administration (NOAA). The actual collection site was approximately 20 miles east of Baltimore. It is believed that this area had a atmosphere with slightly greater visibility than that which was

was reported in Baltimore. Furthermore, the surface observations report horizontal visibility not vertical visibility, which will be slightly greater. Based on this information, a range of 16 to 25 km was established.

In selecting the sampling interval about this range, we consider the change in sensor reaching radiance as a function of visibility. In all the aerosol models of Figures (4.2 – 4.4), we notice that as the visibility gets larger (particularly above 30 km) the variance in the sensor reaching radiance dramatically decreases. From these plots we select the sample points 16, 18, 20, and 25 km, where we have chosen two values below 20 km to capture more of the variability associated with the lower visibilities.

Ground Topography

Ground topography is related to undulation of the terrain. Scene elevation is the altitude of the surface relative to sea level. This parameter, together with the sensor altitude, controls the amount of molecular absorption there is in a column of air from the ground to the sensor.

For this research a scene elevation estimate of 0.015 km (50 feet) was obtained. A second estimate of 0.006 km (20 feet), however, was also obtained from a different source. It is unclear which of the two estimates are correct. Since the values are relatively close to one another (*i.e.*, the difference is not kilometers) their difference will not impact detection results. In general, the scene elevation is relatively easy to obtain since most of the Earth's topography, relative to sea level, has been mapped.

For the FR I test imagery, the sensors altitude was known to be approximately 5146 feet (1.56 km) (see Tables 3.1 and 4.1). To obtain the sensor reaching radiance based on this geometry in MODTRAN, we set the altitude of the sensor (H1) to 1.56

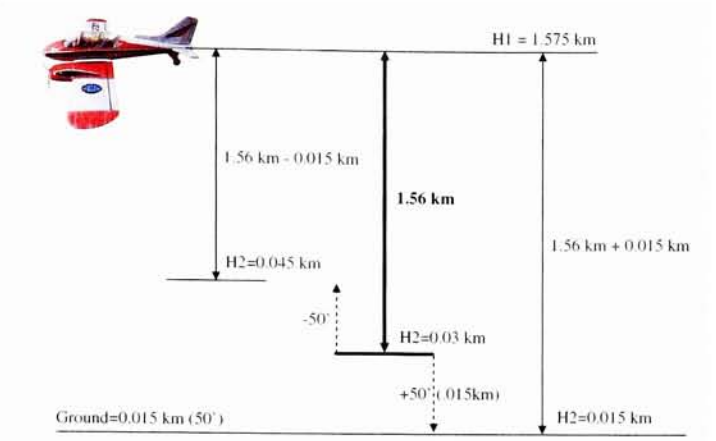


Figure 4.5: Variation in the ground topography parameter as configured in MODTRAN. The sensor height is 1.56 km. We have additionally factored in variation due to scene elevation of ± 50 feet.

km (looking down ANGLE=180 deg) and set the final altitude (H2) to the scene elevation, which was known to be approximately 0.015 km (see Tables 3.1 and 4.1). This can be seen graphically in Figure 4.5. If we wish to introduce variation, we simply change the height of parameter H2. For this research we chose H2 values of 0.015, 0.03, and 0.045 km, while setting H1 to 1.575 km. From Figure 4.5 we can see that these values introduce a variation about the ground height of approximately ±50 feet. This range in variation was selected because of the variation in estimating the scene elevation. There was no ground variation (*i.e*, factoring in major hills or mountains) because it was known at the time of collect that the ground below being imaged was fairly flat.

At a visibility of 20 km the variation in ground topography (with a sensor height of 1.56 km) is relatively small, as can be seen in Figure 4.6. Here we have fixed the visibility to 20 km while using an extinction model for rural aerosols. We then fix the sensors height to 1.56 km and vary the final height (H2). Though we see changes

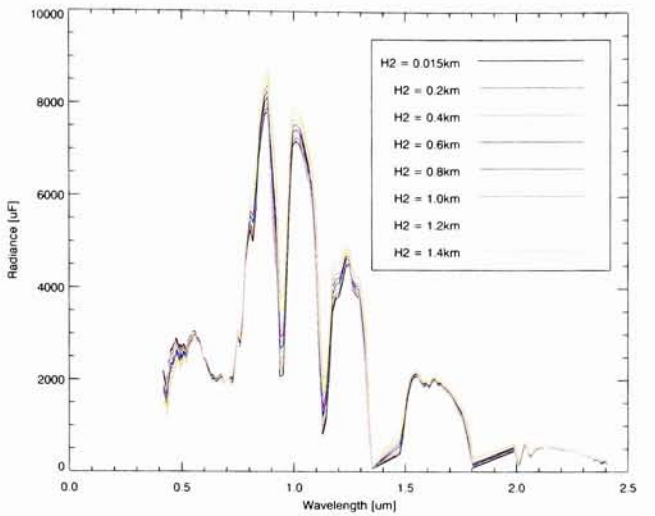


Figure 4.6: Variation in sensor reaching radiance as the ground topography parameter is altered (H2). For this plot we fixed the visibility to be 20 km, used a rural extinction model, and set the sensor height to 1.56 km.

through out the entire spectrum, the largest change occurs in the amount of water absorption around $940\,\mu m$ and $1140\,\mu m$.

Though this variation, for the sensor/scene geometry used in this research, is relatively small, we have presented a method on how to vary the amount of molecular absorption there is in a column of air from the ground to the sensor. The impact of this variation increases with terrain variation and decreased visibility.

WV Scale Factor

As explained in Sec. 3.1.3, water vapor is by far the largest contributor to absorption in the atmosphere. The basis for this type of absorption in our physical models comes from one of the 6 default MODTRAN atmospheric water vapor profiles, which are atmospheric model dependant (*cf.* Figure 3.3).

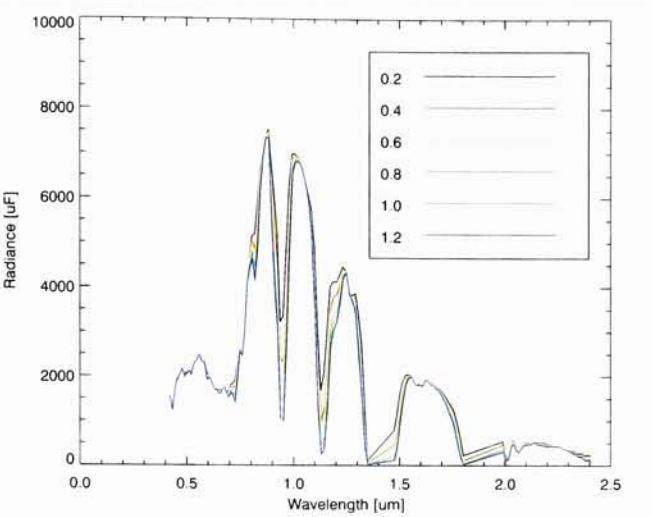


Figure 4.7: Plot showing variation in sensor reaching radiance as a function of water vapor scale factor. Atmospheric model used was that of mid-latitude summer with a visibility of 30 km.

As mentioned in Sec. 3.1.3 the method in which the column water vapor is varied is through use of a water vapor column scaling factor. The sensor-reaching effects of modulating the column water vapor can be seen in Figure 4.7. For this plot, the sensor height (H1) was set to 1.56 km with the final height (H2) set to 0.015 km. We then varied the water vapor scale factor from 0.2 to 1.2, using the mid-latitude summer model atmosphere with a visibility of 30 km. In this plot we notice a significant increase in absorption near the water bands (*i.e.*, 940 nm and 1140 nm) as the WV scale factor is increased.

Ideally, one would like to have a radiosonde profile at the collection site taken at the time of image acquisition. Since radiosonde data was not available, we chose to estimate the integrated column water vapor through use of physically based algorithms. The water vapor retrieval algorithms CIBR and APDA (*cf.* Sec. 3.1.3) were

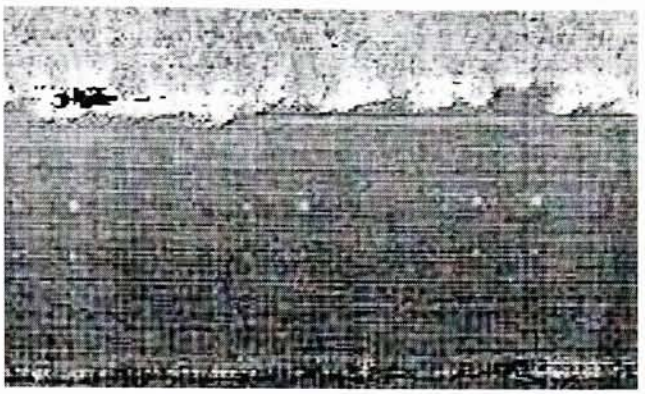


Figure 4.8: Estimation of per pixel total column water vapor using the FLAASH algorithm.

not available at the time of this research. However, as mentioned in Sec. 3.1.3, a by-product of physically based atmospheric compensations algorithms is an integrated column water vapor map. For this research, water vapor maps were obtained from the FLAASH algorithm using a mid-latitude summer model atmosphere and similar scene geometry (*i.e.*, TOD, DOY, sensor height, etc.) as used in this research. The FLAASH water vapor map used can be seen in Figure 4.8.

In order to estimate the range of water vapor values, a histogram of the FLAASH WV map was created in units of $[g/cm^2]$, as can be seen in Figure 4.9. From this we can see that most of the water vapor values fall between 1.5 and 2.0 $[g/cm^2]$, with some of the values being as large as 2.3 $[g/cm^2]$. Based on this, the mean value is approximately 1.8 $[g/cm^2]$. Therefore, we would like a MODTRAN WV scale factor that produces integrated values in this range with a mean integrated value of approximately 1.8 $[g/cm^2]$, though this is a little large, since the histogram is not truly Gaussian. A more realistic value might be around 1.75 $[g/cm^2]$.

To see the effect WV scale factor has on our given geometry (*i.e.*, a sensor height of 1.56 km), a set of MODTRAN runs was established where the WV scale factor

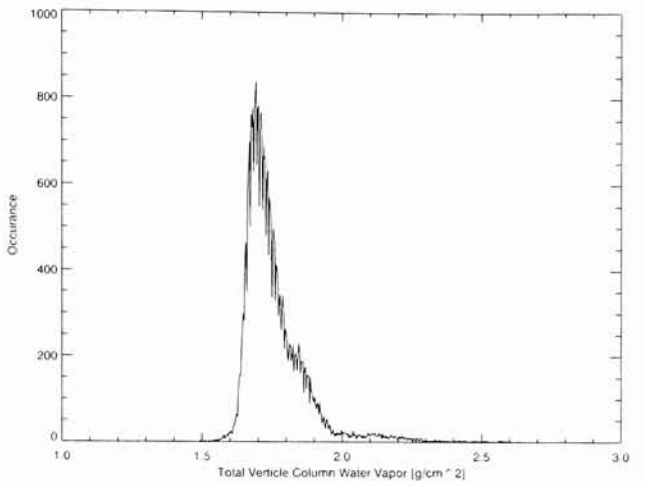


Figure 4.9: Histogram of water vapor map produced by the FLAASH algorithm.

was varied and total vertical column water vapor amount extracted. MODTRAN reports both the line_of_sight path (measured from the sensor to the ground) as well as the vertical_path_to_space (measured from the ground to space along the solar illumination path) integrated water vapor values. The latter of these two (vertical_path_to_space) is what we call total column water vapor and is used to compare against FLAASH derived values.

The results of these runs can be seen in Figure 4.10. As expected, the integrated water vapor value varies linearly with scale factor. That is, for this mid-latitude summer atmospheric profile we have the relation

$$WV_{scalar} = 0.34(TCWV_{mls}) + (\approx 0). \tag{4.1}$$

Therefore, in order to obtain the MODTRAN scale factor, we only need to know the total integrated water vapor amount for the atmospheric WV profile. From this, we can linearly scale to the total water vapor amount to obtain the scale factor.

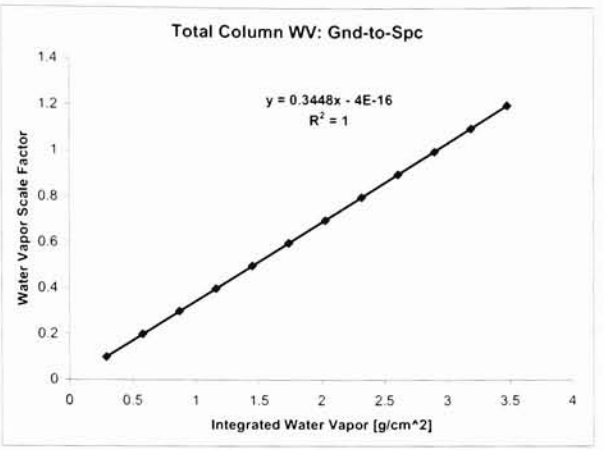


Figure 4.10: Plot showing the linear dependence of water vapor scale factor to total integrated water vapor.

Using this model we can estimate the water vapor scalar for an integrated column value of 1.75 $[g/cm^2]$. This produces a scalar of 0.6. We can also estimate the extreme scalar values for the range which were found to be 0.5 and 0.7. To encompass the full histogram, those WV values in the tails, we extend this range slightly to 0.4 to 0.8. As for sampling in this range we simply sample in an interval of tenths producing the set of values 0.4, 0.5, 0.6, 0.7, 0.8.

4.1.4 Generating a Target Space

Once we have completed our MODTRAN runs we can begin to parse the look up table (LUT) to obtain the terms in Eq. (3.6). The look up table does not contain each variable in Eq. (3.6) but rather groups the terms according to the following,

$$L_s = (L_{drct\_refl} + L_d)r + L_u \tag{4.2}$$

where L_{drct_refl} is the signal from the sun-to-ground-to-sensor. That is,

$$L_{drct\_refl} = E'_s \cos\sigma \, \tau_1 \tau_2 r + L_d \tau_2 r. \tag{4.3}$$

The downwelled term L_d is calculated as the difference between L_{gnd} and L_{drct_refl} where we express L_{gnd} as

$$L_{gnd} = (L_{drct\_refl} + L_d) r. \tag{4.4}$$

To obtain the upwelled or path radiance L_u, the surface albedo in MODTRAN is set to zero so as to eliminate the direct and downwelled terms. This is computed as an additional run during the MODTRAN creation stage. The final output of all the MODTRAN runs is a formatted ASCII file containing the results of the above calculations. The remaining varied parameters are implemented as a post process operation and are described in the next section.

Not included in the above model is radiance due to adjacency effects. These are photons that originate from reflected radiance of the surrounding area, out of the field of view, and are scattered by the atmosphere into the field of view. Since this is similar to a path scattering term, it can cause image blurring, to a lesser extent, and reduce the overall contrast of the image. To explore this effect, a possible average background albedo could be incorporated as a means to calculate path radiance due to adjacency effects. This will be the subject of future research.

Varying Target Orientation and Shape Factor: Fully Exposed Imagery

Once the MODTRAN runs are completed we can implement variations in target orientation angle as well as shape factor values as a post process operation. It is at this stage that we additionally implement spectral resampling, units conversation,

and determination of bad bands, and ultimately band elimination, due to low signal-to-noise and atmospheric water absorption.

For the fully exposed imagery it was known that the targets were placed on relatively flat ground. Therefore, the expected angle of the targets, relative to Sun, should not vary much (*i.e.*, rotation angle should be zero). However, a slight change in illumination ($\pm 10\%$) was introduced which produced a rotation angle range of ± 3 degrees for the given time of day (*i.e.*, zenith was 60 degrees). Therefore, the angles implemented in generating the target space were -3, 0, and +3 degrees. Intermediate angles (*i.e.*, 1 and 2 degrees) were not used because such small rotation angles (and therefore small percentage change in illumination) would not add any additional benefit to the target space in addition to making the target space un-necessarily large. The true impact this sampling has on detection performance is the subject of future work.

In general, the user can either base the targets rotation angle range on knowing something about the terrain and thus how the targets are angled to the Sun, or base the angle range on a desired percentage of illumination gained or lost.

It should be noted that the selection method is a function of the time of day and therefore the zenith angle. For example, if we had a rotation angle of 5 degrees and the zenith angle was 60 degrees, we would expect a 16% decrease in direct illumination onto the target. If, however, the zenith angle was 10 degrees (*i.e.*, different time of day), we would expect a 2% decrease in direct illumination for the same rotation angle.

The shape factor value used for the fully exposed imagery was 0.8. That is, we reduced the downwelled contribution onto the target by 20%. This value was selected based on knowing the layout of the imaged site. As can be seen in Figure

3.11, all of the panels were adjacent to a nearby forested region thus reducing their exposure to the hemisphere above. Based on this, an estimate of approximately 20% was used. For this research, no range was implemented in lieu of future work which will elucidate the impact this downwelled term has on detection results.

Varying Target Orientation and Shape Factor: Fully Concealed Imagery

For the fully concealed imagery, we would like to reduce the direct solar term to almost zero. In a rotation sense, this would have the same effect as rotating the target such that it was 90 degrees from the Sun. Since the zenith angle was known to be 36 degrees, a rotation angle of -54 degrees [*cf.* Eq. (3.4)] would produce a direct term contribution of zero. More than likely, however, there is some amount of direct illumination (propagating through the tree canopy, etc.) on the target. Therefore, we reduce the direct term by approximately 90% rather than 100%. In doing so, we select a rotation angle of -49 degrees which reduces the direct term approximately 90%. We chose not to vary this term since most of the variation is related to the amount of downwelled radiance onto the target not the amount of direct solar radiance.

As previously mentioned, there is more uncertainty in the shape factor term than the direct term. This is due to the fact that we only know the targets are concealed, however, we don't know the level of concealment. Therefore, we seek a range of shape factor values that can potentially encompass this level of concealment. The shape factor values chosen based on the concealed image (*cf.* Figure 3.12) and photographs of the targets under concealment, were 0.3, 0.4, and 0.5. We chose 3 values simply to keep the target spaces for the exposed and concealed data sets the same size. The actual impact variation in this parameter has on detection, will be

investigated in future work.

4.1.5 Summary of Target Space Generation and Calibration Issues

In the previous sections we explicitly outlined the process of creating hundreds of spectra utilizing the atmospheric propagation code, MODTRAN. Additionally, we listed the various parameters required by the user in order to create such spectra. This was followed by sections that differentiated parameters in terms that were fixed (but selected by the user) and ones that were varied over a range. These varied parameters included visibility, ground topography, water vapor scale factor, target orientation (or effective reduction in direct solar radiance), and shape factor. A summary of the varied parameters and their ranges can be seen in Table 4.2. These combinations produced a total of $4 \cdot 3 \cdot 5 \cdot 3 \cdot 1 = 180$ target space vectors for the exposed imagery as well as $4 \cdot 3 \cdot 5 \cdot 1 \cdot 3 = 180$ vectors for the fully concealed imagery. We can see that compared to previous studies [14], the target spaces created in this research are very small. Future work will analyze how growing target spaces (to include more target variability) impacts target detection.

In addition to differentiating fixed from varied parameters, we also explained why the parameter was varied, what value was used, the level of difficulty in knowing the parameter, why the selected range was chosen along with the sampling with in that range. Common to all these parameters is that they are scene dependent. Though we have taken some of the guess work out of the process, there still remains the *quantitative* impact some of these variables have on detection results, though in some sections we have explain what our intuition would lead us to believe. This topic will be addressed in future research.

Table 4.2: Summary of varied parameters for both the fully exposed and concealed images.

	Exposed Image	Concealed Image
Visibility [km]	16, 18, 20, 25	16, 18, 20, 25
GND Topography (H2)† [km]	0.015, 0.030, 0.045	0.015, 0.030, 0.045
WV Scale Factor	0.4, 0.5, 0.6, 0.7, 0.8	0.4, 0.5, 0.6, 0.7, 0.8
Target Orientation‡ [deg]	-3, 0, 3	-49
Change in Target Illumination‡	-9.2%, 0, 8.8%	-89%
Shape Factor	0.8	0.3, 0.4, 0.5

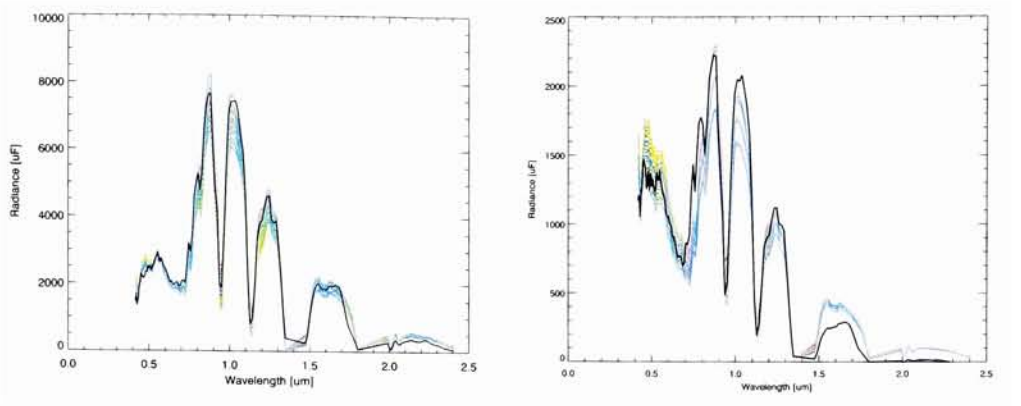
†See Sec. 4.1.3 for details on geometry parameters.

‡ Relative to the target normal.

After generating MODTRAN spectra and implementing the post processing steps (*i.e,* inclusion of target orientation, shape factor, etc.), we can finally generate a target space. Using Eq. (3.6) and the values found in Tables 4.1 and 4.2, along with a measured reflectance of the target, we generated the target spaces shown in Figure 4.11. In both Figure 4.11(a) and (b), we have over plotted an actual image pixel of the F3 target panel. We can see that both target spaces (exposed and concealed) encompass the image derived target pixel rather well. The exception is in SWIR region for the fully concealed case, which we will address below.

Calibration Issues

In Figure 4.11, we notice there are some slight discrepancies in the fit between our model and the image derived target pixel. This lack of fit is more pronounced in the SWIR for the fully concealed target space. Upon initial investigation, this difference was believed to be related to a calibration error associated with pre-1997 HYDICE

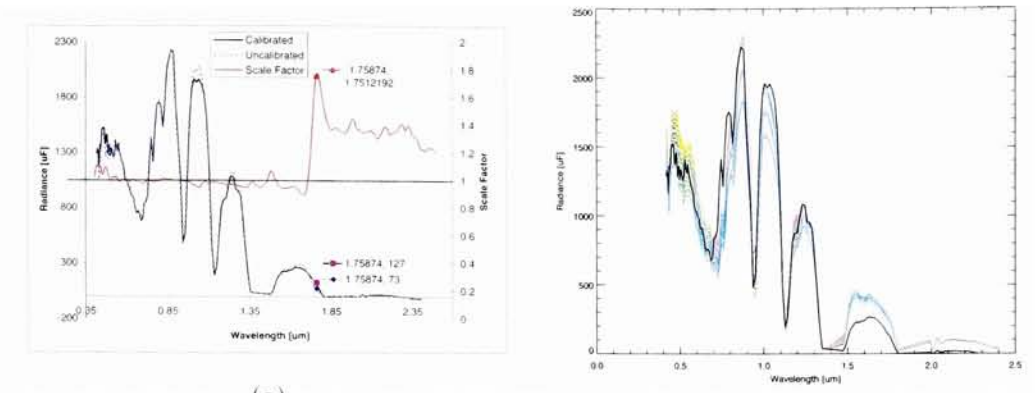


(a) (b)

Figure 4.11: (a) Illustrates a 180 vector target space generated for the fully exposed imagery while (b) shows a 180 vector target space for the fully concealed imagery. Over plotted in both target spaces is an actual image derived F3 target pixel.

imagery. Data of this vintage had a discrepancy due to an error in the absolute calibration of the in-flight calibration bulb and was found to be most pronounced in the 1.7 to 2.5 μm region. The solution involved developing a set of correction coefficients that related 1995 to 1997 FCU radiance data.

It was known at the time of this research that the exposed imagery (run05) already had this correction factor applied. However, upon a second iteration, or review, of the results presented in this research, it was determined that the correction factor was not applied to the fully concealed imagery (run25). In Figure 4.12(a) we see the results of applying such a correction curve to the F3 pixel in the concealed imagery. The correction curve appears as a red line roughly centered around one. Though there are some noticeable changes around 0.464 and 1.04 μm, we can see that the scale factor has not increased our radiance values in the SWIR to a level predicted by the model. As an example of this scaling, we show two values at a



(a) (b)

Figure 4.12: (a) Plot showing absolute calibration correction factor (red line) and the result of multiplying such a correction factor to an F3 pixel in the fully concealed imagery. (b) Shows the same target space in Figure 4.11(b) except with the calibrated F3 pixel over plotted onto the fully concealed target space.

wavelength of 1.75874 μm. We can see that with such small radiance values, the increase is negligible. This leads us to believe that the discrepancy in the SWIR is unrelated to the above mentioned calibration error and needs to be investigated. Still, we go ahead and apply the correction factor to the data knowing that this source of error needs to be addressed in future studies. The fully concealed target space with the *calibrated* F3 pixel over plotted can be seen in Figure 4.12(b). Here we see a slightly better fit to our target space in both the 0.464 and 1.04 μm regions but negligible improvement in the 1.6 to 1.7 μm region.

4.1.6 Target Space Evaluation

One of the goals of this research is to demonstrate that target detection on uncompensated radiance imagery is a possibility. Typically, target detection is per-

formed on compensated imagery where atmospheric compensation algorithms state their perform and/or ability to obtain accurate ground leaving reflectance, in reflectance units. A similar approach can be used here by first modeling a given radiance image pixel $\mathbf{t}_{pix}$ as a linear combination of target space vectors. That is we assume a linear mixing model of the form

$$\mathbf{t}_{pix} = \mathbf{T}_{spc}\,\boldsymbol{\alpha} + \boldsymbol{\epsilon} \tag{4.5}$$

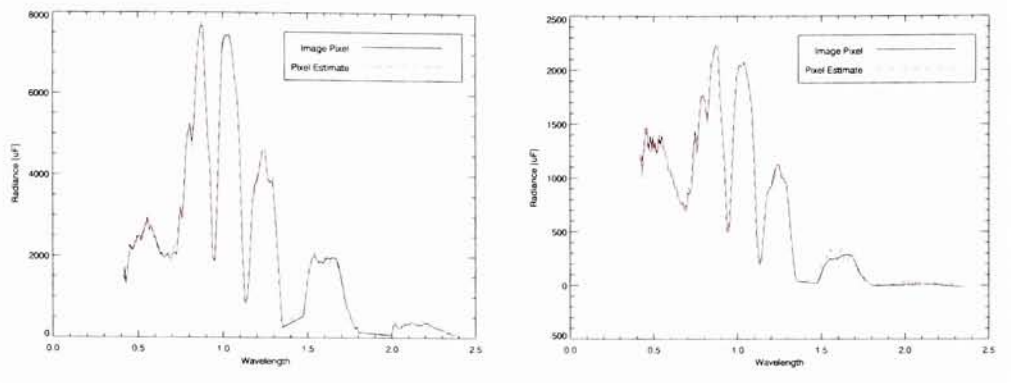
where $\mathbf{T}_{spc}$ is a matrix of target space spectra, $\boldsymbol{\alpha}$ is a vector of weights, and $\boldsymbol{\epsilon}$ is the residual error or model lack of fit. It is this error we wish to minimize. The least squares solution of $\boldsymbol{\alpha}$ that minimizes the error $\boldsymbol{\epsilon} = |\mathbf{t}_{pix} - \mathbf{T}_{spc}\,\boldsymbol{\alpha}|$ is of the form

$$\boldsymbol{\alpha} = \boldsymbol{T}^{\dagger}_{spc}\boldsymbol{t}_{pix} \tag{4.6}$$

where, using a singular value decomposition approach, $\boldsymbol{T}^{\dagger}_{spc} = \mathbf{V}\boldsymbol{\Lambda}^{-1}\mathbf{U}^{T}$. The diagonal matrix $\boldsymbol{\Lambda}^{-1}$ is made up of j singular values, w_j such that $\boldsymbol{\Lambda}^{-1} = [\mathrm{diag}(1/w_j)]$.

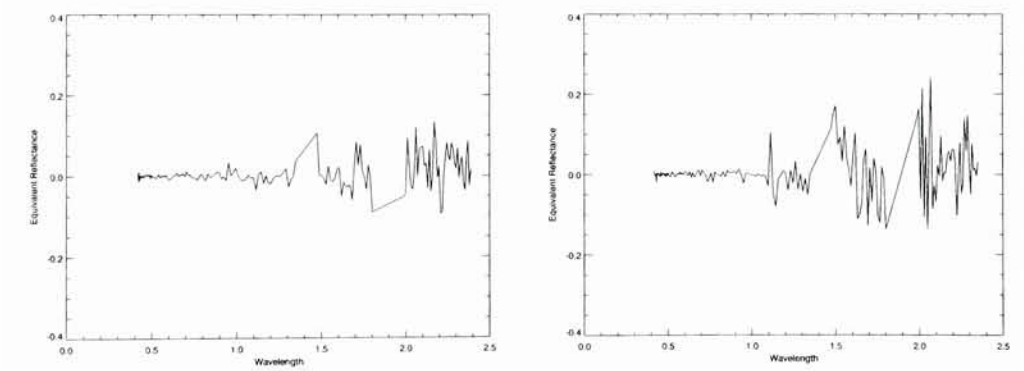
A plot of the image target pixel (in radiance space) and target pixel estimate, for both the exposed and concealed scenes can be seen in Figure 4.13. The estimation fits the actual image pixel well. We can further quantify this fit by computing the difference between the vectors. Additionally, we can convert this quantity from radiance to reflectance by normalizing by the radiance that would be produced had we had a unit reflector (*i.e.*, r=1). That is, we use Eq. (2.49) again setting r=1. Since there are 180 target space vectors, we choose the single vector that was generated using our best guess of the atmosphere (*cf.* Sec. 4.1). The results of this subtraction and normalization can be seen in Figure 4.14 for both the exposed and concealed imagery.

Finally, we can compute an RMS error as a function of wavelength. This is similar to a cumulative RMS error where the last value in the sum would be the



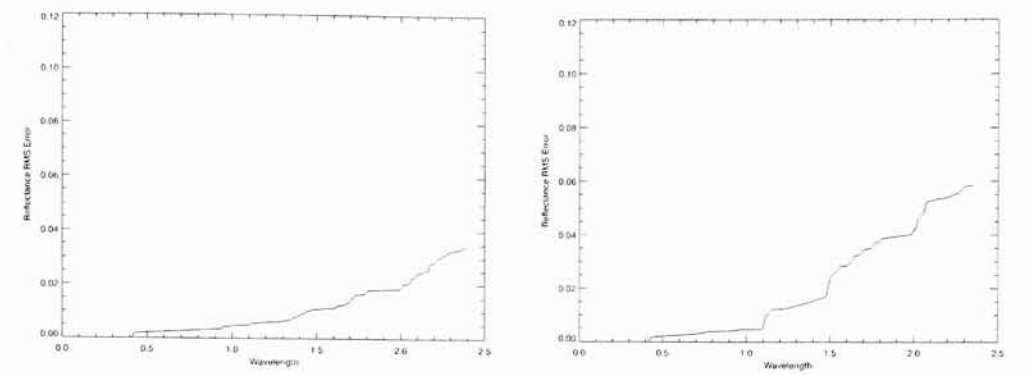
(a) (b)

Figure 4.13: Estimation of target image pixel using physically derived target vectors for the (a) exposed and (b) fully concealed data sets.



(a) (b)

Figure 4.14: Plotting the reflectance normalized difference between the target image pixel and target pixel estimate using physics based modeling for both the (a) exposed and (b) concealed data sets.



(a) (b)

Figure 4.15: Cumulative RMS error as a function of wavelength for both the (a) exposed and (b) concealed data sets. The last value in these plots is the total RMS error.

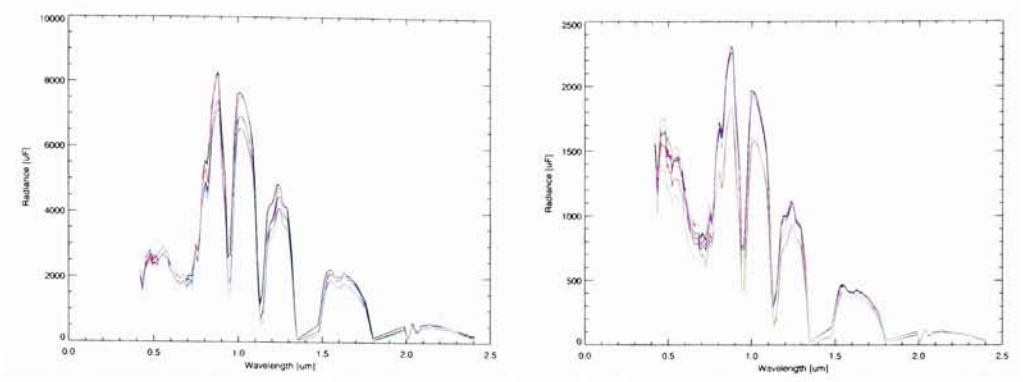
total RMS error. That is

$$RMS_i = \left[\frac{1}{N} \sum_{i=1}^{N} \left(\frac{\boldsymbol{t}_{pix}(i) - \hat{\boldsymbol{t}}_{pix}(i)}{\boldsymbol{t}_{r=1}} \right)^2 \right]^{1/2} \tag{4.7}$$

where N is the total number of bands, $\hat{\boldsymbol{t}}_{pix} = \mathbf{T}_{spc}\,\boldsymbol{\alpha}$, and $\boldsymbol{t}_{r=1}$ is the radiance spectrum when the target has a reflectance of one. This result can be seen in Figure 4.15 where we can see the error in the visible for both the exposed and concealed data sets is below 0.5%. The cumulative error increases as we get to longer wavelengths due to the increased lack of fit between the model and pixel.

4.2 Target and Background Endmembers

In this section we discuss the process of obtaining a geometric description of the target space and background imagery through use of vector algebra. In all cases, we seek out endmembers using the geometric method called the Maximum Distance



(a) (b)

Figure 4.16: Plot showing 7 endmembers extract from the target spaces generated for the (a) fully exposed and (b) fully concealed image data sets.

method (MaxD), as discussed in Sec. 2.2.3. We then discuss a method that helps mitigate against target pixels from becoming potential background endmembers.

4.2.1 Target Endmembers

Endmembers were generated from the physically generated target spaces using MaxD. For the target spaces generated, a total of 7 endmembers were obtained, based on both the fully exposed and concealed data sets (*cf.* Figure 4.16). We selected 7 endmembers based on previous studies [3] where it was determined that utilizing approximately 7 endmembers with around 14-17 background endmembers produced the best detection rates when applied to FR I imagery.

4.2.2 Background Endmembers

When describing an image background using the concept of vector algebra, one runs the risk of selecting a desired image target as an endmembers. If this happens,

then the target signature will be grouped with the background description and will ultimately get suppressed in the detection stage, where we look to suppress the background only. Depending on the overall target-to-background contrast, this may not be that uncommon. In a high contrast situation, where the target pixel(s) stand out from the background, in a geometric sense, it is highly likely that the targets will be selected as endmembers. Since most endmember finding methods locate the minimum convex set using a geometric convex hull approach, the target pixels are sure to be located at one of the vertices of such a hull. On the other hand, if the target has a very low contrast, relative to the background, then it probably won't get selected as a candidate endmember. However, in this situation, if the target/background contrast is too low, then we won't have any chance of finding the target in the first place since it may end up looking like background.

Augmenting the Background Space

To circumvent this dilemma, we seek background endmembers using an augmentation approach [31]. In this approach, we simply augment the background data vectors with the physics derived target space vectors. The idea is that we are trying to mask or shield the actual target image pixels with the physically derived ones. If the shielding process is successful, then the endmember finding routine will locate the physically derived vectors before it finds the actual target image pixel. Since we know which vectors went into the augmentation, we can check for them and remove them accordingly after the endmember selection process. This is a very promising solution to the problem of inadvertently finding target endmembers and is made possible simply because we are using a physically derived target space. Issues related to this technique have not been fully explored in this research because of time

constraints and keeping to relevant subject matter. This is certainly the topic of future research and will be explored heavily.

To further aid in this process, by reducing processing time and maximizing variance as it relates to information content, we can implement our endmember finding routine on augmented data in a dimensionally reduced space. That is, we can apply a principal components or noise adjusted principal components transform to the data. For this research, a PC transform was applied to the augmented data before the endmember routine was applied.

Using the MaxD algorithm, 15 background endmembers for both the exposed and fully concealed data sets were found. This number of endmembers was chosen based on previous studies [3] that showed that a background description using approximately 14 to 17 endmembers produced the best detection rates when applied to the FR I data set. For the exposed imagery, one of the target panels (F2) was found to be an endmember. Since MaxD looks for extrema in the data set, and target F2 was much brighter than any other target, it was not surprising that this panel was found. However, this can be a problem if F2 was to be the desired target. This issue, and possible solution, is further discussed in the paragraph below. For the concealed image none of the hidden panels were found to be endmembers. This also is not surprising since the panels were concealed and the imagery was much "busier" than the exposed image. In the exposed imagery the targets were in the open and are more likely to make up vertices of the convex hull.

It was found that the "masking" process did not always work in shielding the target pixels. This could be due, in part, to calibration and noise issues, which were not factored into the target spaces. It was also observed, however, that when computing endmembers in a reduced space, we were less likely to find the actual

target pixels than had we run the endmember finding routine on the original data set (in radiance space). It should be noted, however, that there were cases in which the target pixel was still found to be an endmember. It is believed that this would not be an issue if we truly captured the target inside the convex hull of Figure 4.11. This issue of masking or shielding the image target pixels with physics based target spaces will be investigated in future work.

4.3 Detection Results For PBosp, SMF, SAM

In this section, we present the results of applying the PBosp detector to the Forest Radiance I, exposed and concealed imagery. As a means of comparison, we additionally apply the (stochastic) Spectral Matched Filter (SMF) and Spectral Angle Mapper (SAM) to the data sets as well. The results are broken up into two categories pertaining to the exposed and fully concealed imagery. Within each of these categories, we present the raw 2-dimensional test statistic images, in pairs, such that the first image is the original test statistic, byte scaled and the second is a version with a threshold applied. This layout of the image pairs, as well as order of appearance, is illustrated in Table 4.3. At the end of each section, we present the results of generating ROC curves based on both the exposed and fully concealed data sets.

4.3.1 FR I Exposed Imagery Results

The first data set explored was from the FR I exposed collection. Each algorithm was applied to this data and and results were obtained both qualitatively and quantitatively. In order to more readily interpret visual results, we include the exposed target mask of Figure 3.14 again here in Figure 4.17.

Table 4.3: Summary of test statistic images for both exposed and concealed data sets

Test Static Image Result	Test Statistic Threshold Image Result
SAM	SAM w/ threshold
SMF	SMF w/ threshold
PBosp Using bv†	PBosp Using bv† w/ threshold
PBosp Using Mean‡	PBosp Using Mean‡ w/ threshold

†Using 7 basis vectors for $\mathbf{T}$

‡Using single mean target space vector for $\mathbf{T} \Rightarrow \mathbf{t}$

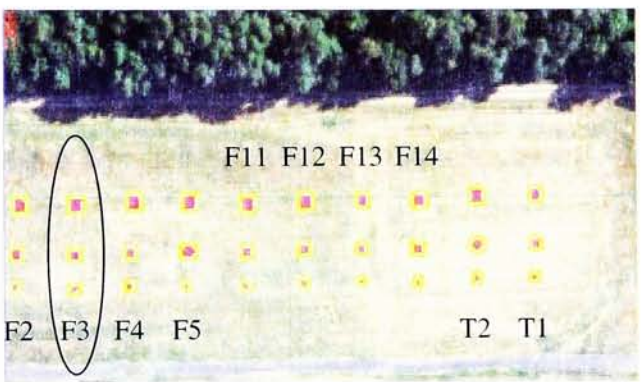


Figure 4.17: Target labeled FR I exposed image. The targets are in columns, above and below each label.

SAM - Exposed

The first algorithm to be applied was the spectral angle mapper (*cf.* Sec. 2.3.1.) This algorithm was applied to ground-leaving *reflectance* imagery where the Empirical Line Method (ELM) was used to perform atmospheric compensation. We selected reflectance imagery (over radiance imagery) because this is how the algorithm is typically applied. However, it should be noted that this "research grade" reflectance

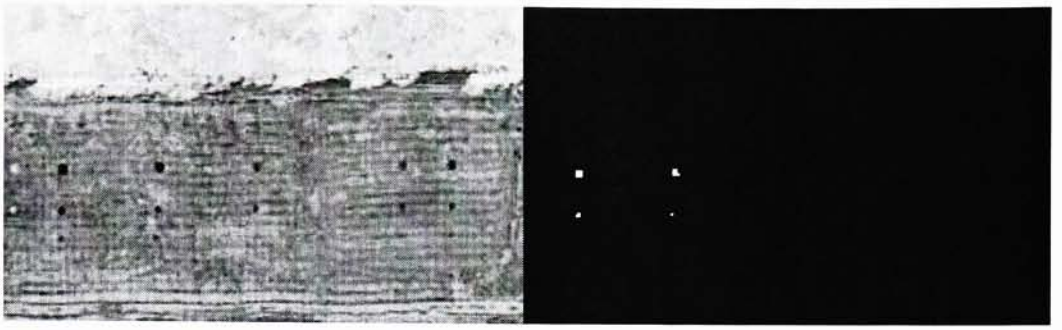
imagery is not what the analysis typically has available. Therefore, we should expect an upper performance bound as well as overly optimistic results for the SAM and SMF algorithms. The target spectral reflectance (F3) was obtained from a field spectrometer and resampled to the 170 bands used throughout this research.

The test statistic image produced by this detector can be seen in Figure 4.18. For the SAM algorithm, the expected output is an angle measure in radians, where the smaller the angle the closer the test pixel in question is to the measured reflectance spectrum. This map of byte scaled angles can be seen in Figure 4.18(a) where small angles are seen as dark pixels. We can see that there are many dark pixels on targets F3, F5, and T2. Some of the other targets, like F12 and T1, seem to have slightly larger angles but not nearly as large as the forested region or background grass.

Though the byte scaled test statistic image is useful to visually assess, it does not tell us anything about the top "scores", or in the case of SAM, the top smallest angles. We proceed in this direction by only displaying the top 20 (smallest) angles. This was achieved by establishing a threshold value of 0.157 radians (8 degrees). The resultant image can be seen in Figure 4.18(b). From this, we can see that the top 20 scores are associated with targets F3 and F5. This comes as no surprise since previous studies (*cf.* Sec. 3.3) showed that targets F3 and F5 were very similar in spectral character.

SMF - Exposed

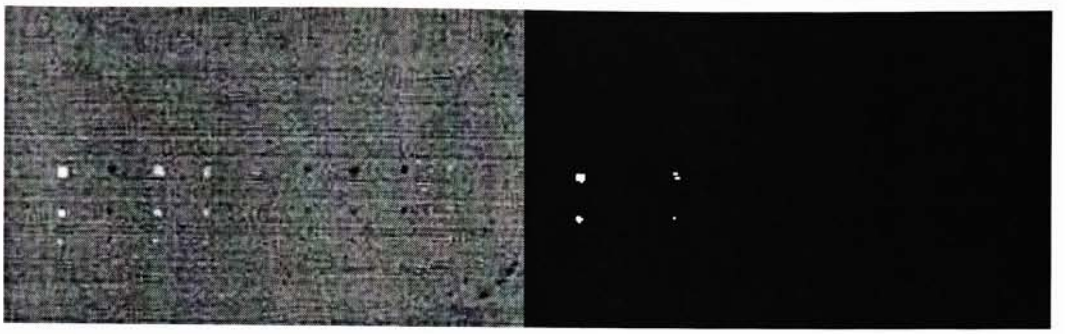
We then applied ENVI's build-in Spectral Matched Filter (SMF) algorithm the the exposed image. This algorithm was also applied to ground leaving reflectance imagery (compensated using the ELM algorithm). Again, with the thought that this is how the user would typically apply this algorithm to hyperspectral data.



(a) (b)

Figure 4.18: Results after applying the SAM algorithm to the FR I exposed data set. The raw test statistic image (a) (map of angles) can be seen next to a version with a threshold applied (b). For the SAM algorithm low values (dark pixels) are equivalent to the most probable detects.

The test statistic results for this algorithm can be seen in Figure 4.19. Unlike the SAM algorithm, the SMF produces *large* values when the test pixel in question is similar to the target reflectance spectrum. Additionally, this detector is stochastic in nature and utilizes a covariance estimated from the entire image. The results can be seen in Figure 4.19(a) where we have slightly different results, in the ranking of targets, than the SAM algorithm. Here we still see that the most probably detects are from the F3 and F5 panels, but we also see that the remaining panels didn't score as high as when the SAM algorithm was applied. Additionally, we notice that the background has been uniformly suppressed. When we proceed to examine the top 20 detects [*cf.* Figure 4.19(b)], we see that the detections are still related to targets F3 and F5. This image was achieved by using a threshold value of 0.054.



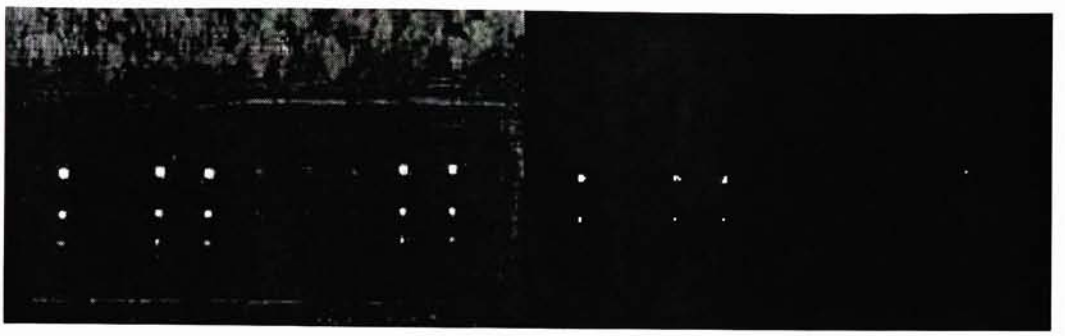
(a) (b)

Figure 4.19: Results after applying the SMF algorithm to the FR I exposed data set. The raw test statistic image (a) can be seen next to a version with a threshold applied (b). For the SMF algorithm high values (bright pixels) are equivalent to the most probable detects.

PBosp - Exposed

The last detector applied to the FR I exposed imagery was the Physics Based Orthogonal Subspace Projection (PBosp) algorithm. This geometric detector was applied to, not reflectance, but sensor-reaching radiance imagery. We start by applying the detector using 7 basis vectors found using the MaxD algorithm. Again, we select 7 based on previous studies that showed approximately 7 target space basis vectors produced maximal results when applied to FR I imagery [3].

The PBosp algorithm, like the SMF, also produces large values that can be associated with highly probable targets. The resultant test statistic image can be seen in Figure 4.20(a). Here we see that the detector has picked up on targets F3, F5, F11, T1 and T2. Again this is not that surprising knowing that the spectral character of these panels is very similar. It does, however, perform poorer than the SAM and SMF algorithms which were applied to research grade reflectance imagery. We can further examine the top 20 detects in order to obtain a visual ranking, as

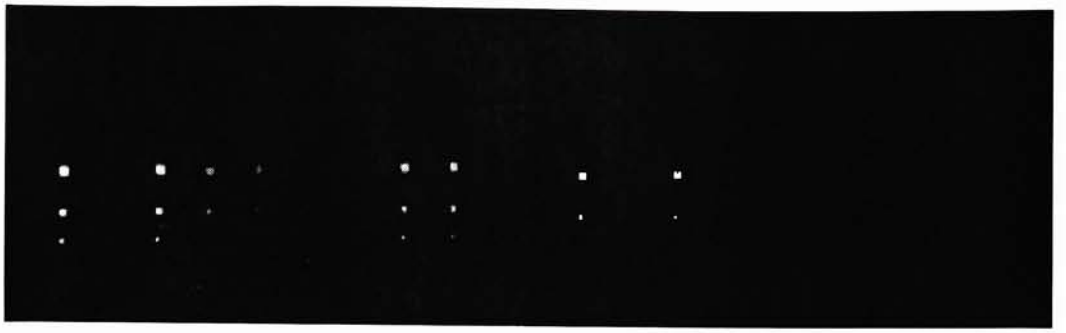


(a) (b)

Figure 4.20: Results after applying the PBosp algorithm, using 7 basis vectors, to the FR I exposed data set. The raw test statistic image (a) can be seen next to a version with a threshold applied (b). For the PBosp algorithm high values or abundances (bright pixels) are equivalent to the most probable detects.

can be seen in Figure 4.20(b). This image was achieved by using a threshold of 0.65. This threshold value, unlike that of SAM and the SMF, is loosely related to target abundance. Here, we notice that most of the detects are indeed associated with target F3 but there are still false detections from targets F5, F11 and one pixel from target T2.

The reason for this may be linked to the size or breadth of the target space. Since these panels all have similar spectral character, they can easily manifest themselves in a target space that is built on "combinations" of how one of the signatures (F3) is to appear at the sensor, though this has not been quantitatively evaluated. The question still remains as to how varied can the target space be before we encounter "crosstalk" issues, as it relates to false alarming on similar targets to that of F3. The target space built in this research was a direct reflection of the scene and atmospheric conditions at the time of collection and should be of "optimal" size. The exact reason for this potential crosstalk is the topic of on-going work.



(a) (b)

Figure 4.21: Results after applying the PBosp algorithm, using the mean target space vector, to the FR I exposed data set. The raw test statistic image (a) can be seen next to a version with a threshold applied (b). For the PSosp algorithm high values or abundances (bright pixels) are equivalent to the most probable detects.

To test the theory of the target space being all to encompassing, the PBosp detector was applied to exposed imagery using only the mean target space vector. This result can be seen in Figure 4.21 where we see improvement over using 7 basis vectors. The test statistic image of Figure 4.21(a) shows the largest amount of background suppression, when compared to SAM, SMF, and the PBosp detector using 7 basis vectors. We also see that high abundances are still coming from targets F5, T1 and T2 with abundances related to F11 greatly reduced. When we threshold this image (using a value of 0.51) to view the top 20 abundances, we can see that targets F11 and T1 have been suppressed while targets F3 and F5 remain. This result is comparable to that found using SAM and SMF on the research grade reflectance imagery.

ROC Curves - Exposed

Quantitative results were obtained using Receiver Operating Characteristic (ROC) curves. These curves were based on the FR I exposed truth mask discussed in Sec. 3.3.3. That is, the mask used, shields or covers up the other man-made targets in the scene. The Detection Rate (DR) for the exposed scene ROC curves is based on trying to identify all 35 full and sub pixel F3 targets (where DR=1 when all 35 targets have been found). The False Alarm Rate (FAR) is related to how many false alarms we incur while trying to find target pixels. For the exposed image, the total number of image pixels was 37,500 such that FAR=1 when we have $37,500 \backslash 37,500$. The curve itself is generated by ordering all 37,500 detection values. We then, starting with the largest detection value, check to see if the value is related to a target or background pixel while keeping tract of detects and incurred false alarms. In an ideal case, the first 35 detection values (for target F3) would be related to the image target.

The performance of each detector can be seen in Figure 4.22. In Figure 4.22(a), we have "zoomed in" on the linear FAR scale such that the maximum value is $1 \cdot 10^{-2}$ which, for this data set, is equivalent to false alarming on 370 pixels out of 37,500. The ROC curves show that SAM and PBosp_bv (using 7 basis vectors) finds about 60% of the target pixels with out any false alarms, while the SMF detector finds about 85% of the targets, also with out any false alarms. The PBosp_mean (using the mean target space vector), however, locates almost 95% of the target pixels with out any false alarms. After this point the curves are fairly flat which means the detectors are incurring FA's while not finding many more targets.

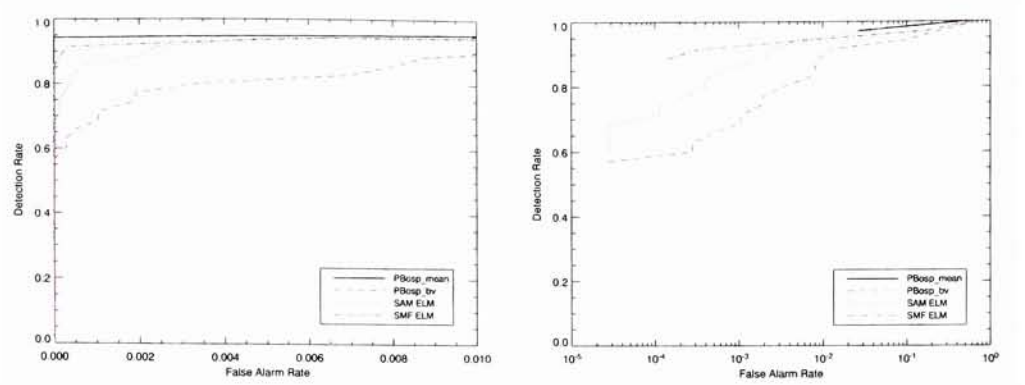
According to the *visual images* discussed earlier, the SMF detector seemed to perform the best, at least when looking at the top 20 detection values. The ROC

curve of Figure 4.22, however, tells a different story because we are now factoring in all target pixels. In actuality, after the first 20 detection values, the SMF starts to false alarm on more background than the PBosp_mean detector does, as the ROC curve illustrates.

Perhaps a more intuitive look at the behavior of the detectors is to plot the ROC curves on a log scale, as can be seen in Figure 4.22(b). This type of plot enables us to view the behavior of the entire ROC curve on a compact scale. Since we only have 37,500 image pixels, the lowest FAR we can obtain is $1\backslash 37,500$ or approximately $2 \cdot 10^{-5}$. The plot shows similar trends to its linear equivalent except now we can also see where the first false alarm occurs for each detector. For example, as mentioned before, the SMF found about 85% of the target pixels with zero FA's. From the log ROC plot we can see that the first FA occurs around $1.5 \cdot 10^{-4}$. For the PBosp_mean detector we see the first FA does not occur until $2.5 \cdot 10^{-2}$. This means it found 33 target pixels with no FA's but then did not find the 34 for another 937 test statistic values. To find the last target pixel, the algorithm had to go through over half the image pixels (result not shown).

Detection: ATREM and FLAASH with SAM and SMF - Exposed

The previous results using the SAM and SMF algorithms, as applied to ELM compensated imagery, assumed one had access to calibration panels in the scene of interest. In a real world situation, however, this is rarely the case. In this event, one might resort to radiation transfer models to perform atmospheric compensation. The FLAASH and ATREM algorithms are two such models that were analyzed, in conjunction with the SAM and SMF algorithms, as to their impact on detection. That is, the FLAASH and ATREM algorithms were both applied to the FR1 im-



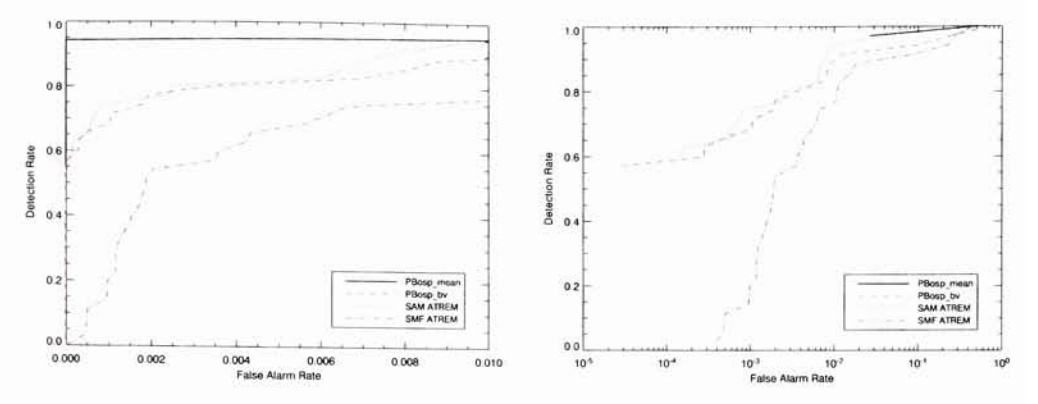
(a) (b)

Figure 4.22: ROC curves for the SAM, SMF, PBosp algorithms, as applied to the FR I exposed data set. The SAM and SMF algorithms used ELM data. Shown are results plotted using (a) linear and (b) log scales. For the log scale in (b), the curves start from the left when the detector encounters the first false alarm (FA). That is, since there is no "zero", the curves appear (or start) when the detector encounters a FA. For the *PBosp_mean* case, the first FA is encountered around $2.5 \cdot 10^{-2}$.

agery producing an estimate of ground leaving reflectance. These cubes were then used with both the SAM and SMF detection algorithms. It should be noted that the cubes were not processed (in house) as part of this research but rather obtained from MIT's Lincoln Laboratory, where processing was performed.

Results using ATREM compensated imagery can be seen in Figure 4.23. Here, we have re-plotted the PBosp results from Figure 4.22 for comparison purposes. We can see that the performance of both the SAM and SMF algorithms has decreased. This is not surprising. The radiation transfer models usually don't compensate as well as the ideal-piecewise ELM approach using known calibration panels. The SAM algorithm now performs similar to the PBosp_bv case while the SMF shows a much sharper decrease in overall performance.

Another data set, compensated using the FLAASH algorithm, was also analyzed



(a) (b)

Figure 4.23: ROC curves for the SAM, SMF, PBosp algorithms, as applied to the FR I exposed data set. The SAM and SMF algorithms used ATREM data. Shown are results plotted using (a) linear and (b) log scales. For the log scale in (b), the curves start from the left when the detector encounters the first false alarm (FA).

in terms of its impact on detection. These results can be seen in Figure 4.24. Again, we have re-plotted the PBosp results from Figure 4.22 for comparison purposes. Here, we see that the SAM algorithm exhibits a similar decrease in performance to that using ATREM. However, we see an over all *increase* in performance when using the SMF algorithm, which is a much difference result than that found in Figure 4.23. In general, we would have expected some level of decreased performance, as was shown with the SAM algorithm. What is questionable is the fact that, using the SMF algorithm, we saw a dramatic decrease in performance when using ATREM but an increase in performance using FLAASH. Since these atmospheric compensation algorithms are fairly similar, we would expect similar trends. Therefore, the results using FLAASH maybe in question and should be reevaluated.

Overall, we can see that the results using the SAM and SMF algorithms are clearly atmospheric compensation dependant. More specifically, we see that the

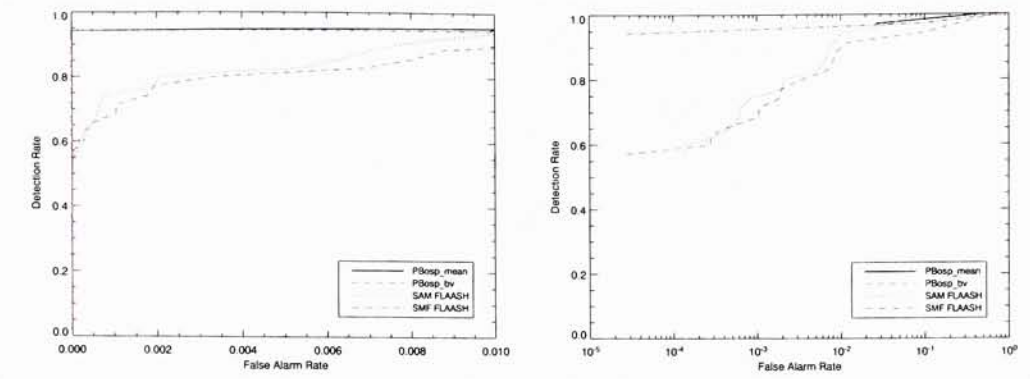


Figure 4.24: ROC curves for the SAM, SMF, PBosp algorithms, as applied to the FR I exposed data set. The SAM and SMF algorithms used FLAASH data. Shown are results plotted using (a) linear and (b) log scales. For the log scale in (b), the curves start from the left when the detector encounters the first false alarm (FA). That is, since there is no "zero", the curves appear (or start) when the detector encounters a FA.

SMF does fairly well, with the exception of the result using ATREM data. This might not be a surprise considering the statistical nature of the test scene. That is, the SMF uses a covariance for its normalization. Therefore, it expects data that adhere to Gaussian statistics. Most scenes do not exhibit normal behavior, however, the exposed test scene used in this research is fairly homogenous and uniform. That is, it does not exhibit large spatial/spectral variation, such as that found in an urban setting. To explore the true behavior of the SMF detector, and its assumptions, one should test the algorithm on larger varying urban-type data.

4.3.2 FR I Concealed Imagery Results

To rigorously test the performance of the SAM, SMF, and PBosp algorithms, we utilized the FR I fully concealed data set where the same targets as in the previous

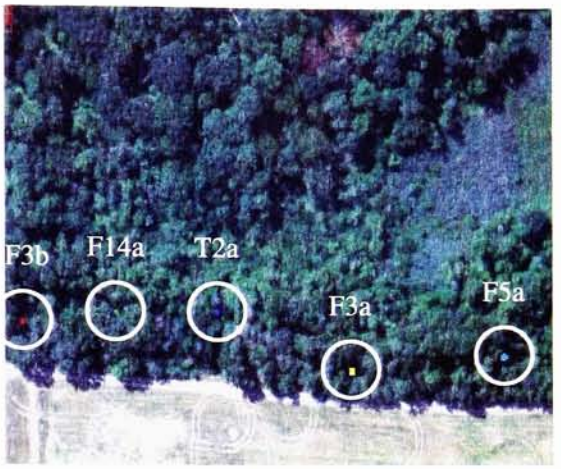


Figure 4.25: Target labeled sub section (250 × 300 pixels) of the larger FR I concealed image.

section were placed in full concealment by moving them to an adjacent forested area. The SAM and SMF algorithms were tested on data that was compensated using ELM, where known calibration panels were used. Unlike the previous section, here, we did not evaluate the SAM and SMF algorithms using the ATREM and FLAASH compensation algorithms due to unavailability of the data.

The term "fully concealed" implies, at the very least, the targets did not have any direct solar irradiance on them. In order to more readily interpret visual results, we include the concealed target mask of Figure 3.15 again here in Figure 4.25.

SAM - Concealed

The first detector to be applied to the concealed imagery was the SAM algorithm. Again, the detector was applied to ground leaving reflectance imagery which was compensated using the Empirical Line Method (ELM). The target we wish to find is the cotton panel designated F3. In the truth image of Figure 4.25 we see multiple

target designations for the F3 panel. These different designations refer to the panels size. All targets with the suffix "a" are 3×3 meters in size while those with the suffix "b" are 2×2 meters in size.

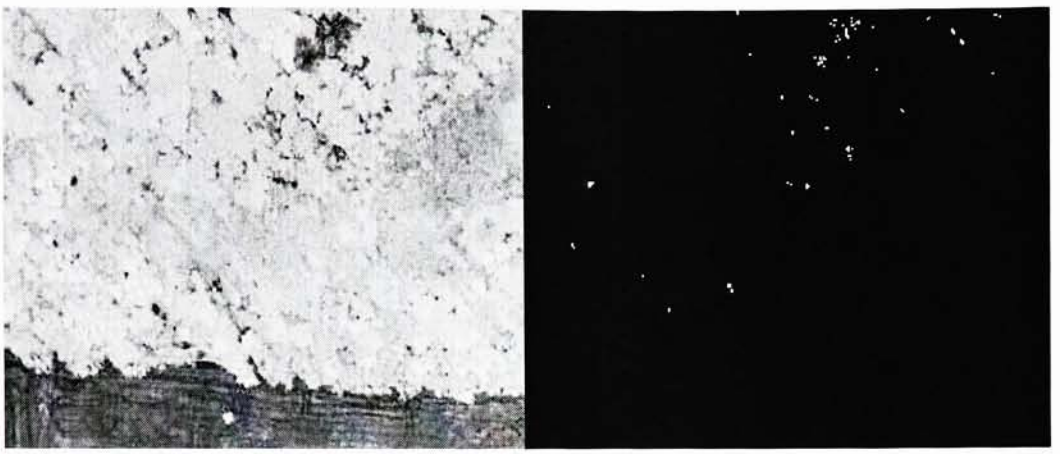
The byte scaled test statistic image produced by the SAM algorithm can be seen in Figure 4.26(a) where dark pixels represent smaller angles. This result is similar to when SAM was applied to the exposed imagery in that the forested region produces overall larger angles than than the grass region. Since the targets are so small and hard to see in this image, we go ahead and set a threshold of 0.2 radians (11 degrees) to view the top 100 test statistic values. This result can be seen in Figure 4.26(b).

When comparing the thresholded result to the concealed truth image, we notice that there are no detections related to any of the target panels, even after 100 false alarms. The reason for this is that the target signature has significantly changed due to spectral illumination effects and computing a simple angle between two vectors is not enough to locate the signature in the image.

Though, to first order, the SAM algorithm is "invariant" to magnitude or brightness changes, the spectral change in illumination is enough to offset this algorithm. Since the panel is in the shade (*i.e.,* no direct solar illumination on it), the flux upon it should be more from the sky dome (downwelled radiance) than the direct Sun (not considering adjacency effects). This is the concept of the invariant method and was discussed in detail in Sec. 2.4.1.

SMF - Concealed

The next detector applied to the concealed imagery was ENVI's SMF algorithm. It was also applied to the ground leaving reflectance imagery, compensated with ELM. The results of this algorithm can be seen in Figure 4.27(a) where high values (bright



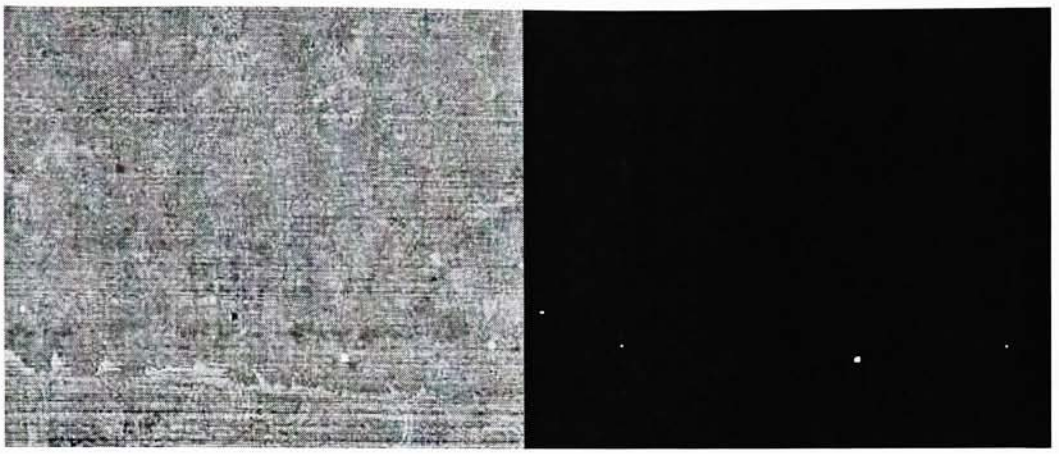
(a) (b)

Figure 4.26: Results after applying the SAM algorithm to the FR I concealed data set. The raw test statistic image (a) (map of angles) can be seen next to a version with a threshold applied (b). For the SAM algorithm low values (dark pixels) are equivalent to the most probable detects.

pixels) are equivalent to the most probable detects. We can see that the SMF, unlike the SAM algorithm, has uniformly suppressed the background. In addition, there are bright pixels in regions where potential targets are located.

We can evaluate this more by looking at the top 12 test statistic values. The image locations for these can be seen in Figure 4.27(b). To produce this image a threshold value of 0.0255 was applied. This image clearly shows the target panel F3a as well as F3b and F5. The SMF false alarmed on a pixel that is located in between targets F3a and F3b. This result is much better than that produced using the SAM algorithm.

Unlike the SAM algorithm, SMF computes an image wide covariance estimate when generating the test statistic [*cf.* Eq. (2.68)]. The algorithm is similar to the statistical distance except that it compares the distance a test pixel is from the



(a) (b)

Figure 4.27: Results after applying the SMF algorithm to the FR I concealed data set. The raw test statistic image (a) can be seen next to a version with a threshold applied (b). For the SMF algorithm high values (bright pixels) are equivalent to the most probable detects.

background mean to the distance the target spectrum is from the same background mean, all normalized by the image wide covariance. The impact the covariance has on the detection is beyond the scope of this research but will be investigated in future work.

PBosp - Concealed

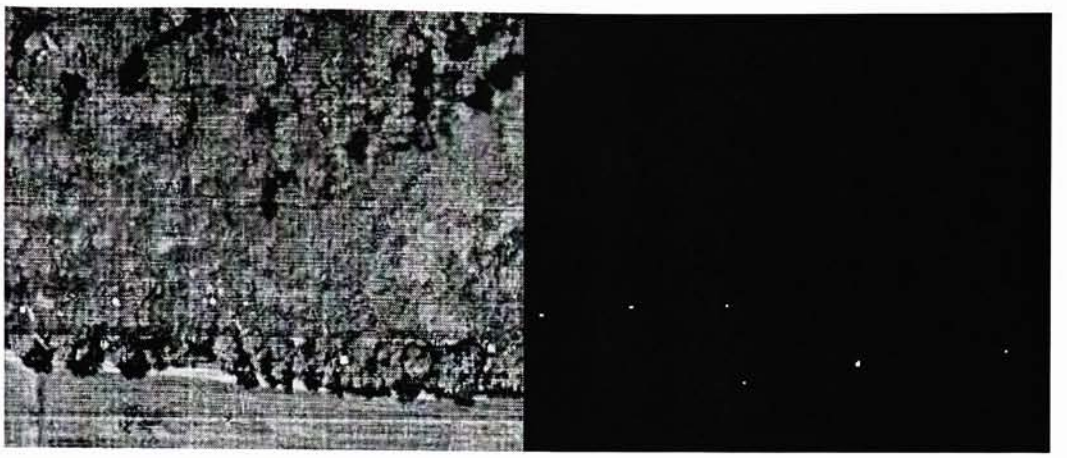
The last detector applied to the FR I concealed imagery was the PBosp algorithm. Once again, we applied the detector to sensor-reaching radiance imagery. As in the exposed image case, we model the target space with 7 basis vectors, computed using the MaxD algorithm. The results of applying the PBosp algorithm, using 7 basis vectors can be seen in Figure 4.28.

The test statistsic (or abundance-like character) image of Figure 4.28(a) shows

more variation in the suppressed background than was seen using the SMF. This maybe due to the number of endmembers along with where they were selected in the image. In a previous study of background selection methods [3], the authors used an FR I image that consisted of a fairly homogenous region of grass. The concealed image used in this research has both grass as well as a busier region of forest and shadow-like regions. The effect of basis vector selection on this data, should also be analyzed and could be the subject of future work.

The test statistic map of Figure 4.28(a), like that of the SMF, shows bright pixels, associated with detects, in regions where there are known targets. If we look at the top 12 detection values, using a threshold of 0.43, we see a similar result to that produced by the SMF. This can be seen in Figure 4.28(b). Using this level of threshold produces detects related to targets F3a, F3b, F5 as well as F14 and a false alarm located between panels F14 and F3. We have essentially found all 5 target panel with a false alarm. This is not surprising since we stated in Sec. 3.3.3 that the panels were all very similar. It should be noted that this method of assessment, viewing the top N-test statistic values, is qualitative in nature. It says nothing about false alarms incurred when looking for *all* the target pixels, which is the focus of the ROC curve.

In order to stay consistent with the exposed imagery test cases, we applied the PBosp detector to the concealed imagery using only the mean vector from the target space. This result is shown in Figure 4.29. Looking at the test statistic image of Figure 4.29(a), we can see that the overall background (both trees and grass) is darker or appears to be more suppressed than when the SAM or SMF detectors were used. This may also be due to the fact that using the mean target space vector in the detector has made the target pixels more likely to stand out. This also tells us

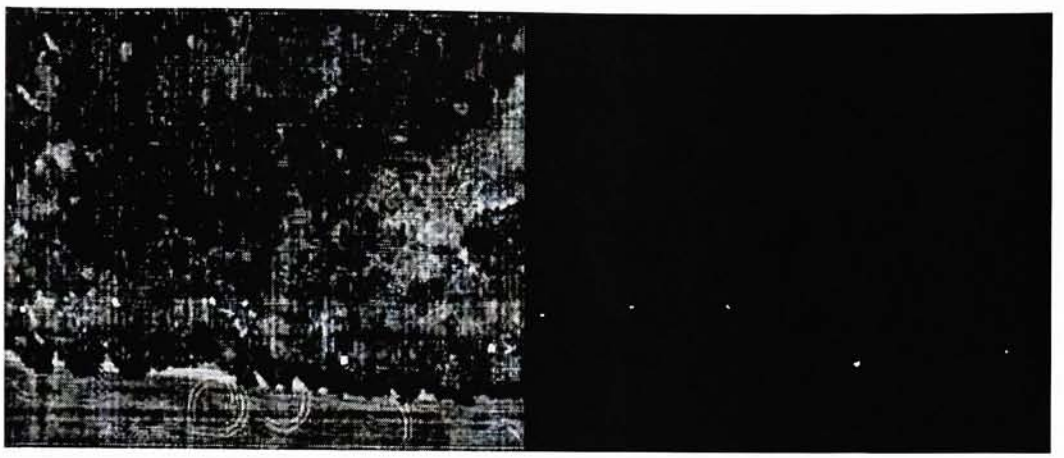


(a) (b)

Figure 4.28: Results after applying the PBosp algorithm, using 7 basis vectors, to the FR I concealed data set. The raw test statistic image (a) can be seen next to a version with a threshold applied (b). For the PBosp algorithm high values or abundances (bright pixels) are equivalent to the most probable detects.

that the target space is not as close (*i.e*, more separation) to the background as in the case when we used 7 basis vectors. This overall trend in using the mean vector versus 7 basis vectors is similar to what was found when evaluating the exposed imagery case.

If we proceed to threshold this image looking at only the top 12 test statistic values (or abundances), we get the result of Figure 4.29(b). To obtain this image a threshold value of 0.391 was applied. The results are similar to that obtained using 7 basis vectors except the false alarm has been removed. Again, we have identified all 5 target panels with the bulk of the detection values being related to target panel F3.



(a) (b)

Figure 4.29: Results after applying the PBosp algorithm, using the mean target space vector, to the FR I concealed data set. The raw test statistic image (a) can be seen next to a version with a threshold applied (b). For the PSosp algorithm high values or abundances (bright pixels) are equivalent to the most probable detects.

ROC Curves - Concealed

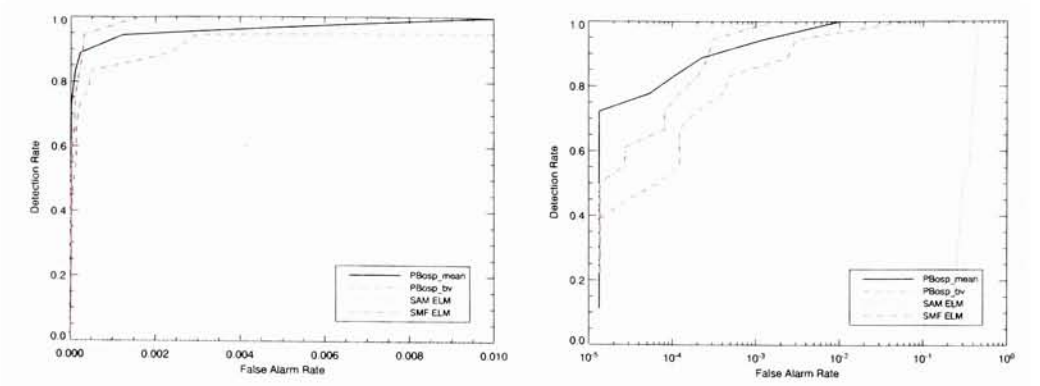
Quantitative results for the fully concealed imagery was generated using ROC curves. These curves were based on the truth masks discussed in Sec. 3.3.3. That is the truth mask used, covers up or masks out the other man-made objects in the scene. The detection rate for the concealed imagery is based on trying to identify all 18 F3 target pixels (12 pixels for F3a and 6 pixels for F3b). In creating the truth mask for this image, there was no attempt made at trying to separate full from sub pixels. The masks simply encompassed the most probably pixels (through visual analysis) as they related to a particular target of interest. The total number of pixels for this image was 75,000 which produces a minimum FAR of $1\backslash 75,000$ or approximately $1 \cdot 10^{-5}$. This is not much lower than was found when using the exposed imagery.

The performance of each detector can be seen in Figure 4.30. In Figure 4.30(a)

we have the same linear scale used when the exposed imagery was evaluated. Unlike results obtained in the exposed case, all the detectors (except for SAM) false alarm rather quickly, after finding about 20% of the target pixels. The PBosp detector using 7 basis vectors found about 40% of the targets (with a few FA's) while the SMF found about 60% of the targets, with approximately the same number of FA's. On the other hand, the PBosp detector using the mean target space vector found aoubt 70% of the targets with a small number of FA's. The SAM algorithm shows up as a line at the bottom of the graph because, in this range, it did not find any target pixels. This shows that the PBosp detector, using the mean target space vector, generally outperformed all the detectors. The SMF also performed reasonably well and found the 17th target pixel before the PBosp detector did, using the mean vector.

The same analysis can be performed by looking at Figure 4.30(b) where we have plotted the results on a log scale. Here we see that the SAM algorithm does not find a target pixel until after 15,000 test statistic values. We also see that the SMF finds the last 2 target pixels before any other algorithm, including the PBosp, using the mean vector. However, in general, the PBosp, using the mean vector, found most of the target pixels in the concealed image incurring the fewest FA's.

One last comment is on the operational aspect of each algorithm. To apply the SAM and SMF algorithms, as we have seen, we usually work in a reflectance domain where target spectra come from a spectral library, for example. In a real-time operational situation, this compensation step can become burdensome and impractical. However, the PBosp approach, using radiance target spaces, can work in this situation since the compensation step is not necessary.



(a) (b)

Figure 4.30: ROC curves for the SAM, SMF, PBosp algorithms, as applied to the FR I concealed data set. Shown are results plotted using (a) linear and (b) log scales.

4.4 Comparison of PB-SIFT and MTMF

In this section we take the PBosp detector and combine it with our Structured Infeasibility Projector (SIP) to generate 2 dimensional decision spaces. Together, this algorithm is called the Physics Based-Structured InFeasibility Target-detector (PB-SIFT). All the results, as they relate to the PBosp detector, have been previously presented. Therefore, this section starts off by looking at the results of the infeasibility projections. We compare this approach to ENVI's Mixture Tuned Matched Filter (MTMF) [5] which incorporates the Spectral Matched Filter (SMF) previously discussed and an infeasibility metric, which we will simply term INF, for analysis purposes. Both algorithms generate 2 dimensional decision spaces where results are compared on a visual basis.

4.4.1 Noise Estimate and MNF Transform

In order to obtain the INF images in ENVI, one needs to run the MTMF algorithm. The MTMF requires the input image be MNF transformed. However, the MNF procedure requires an image noise estimate. In an ideal case, a sensor noise frame (cube) would be available to the user to use as input to the algorithm. However, for this research no external HYDICE noise data was available. In this event, an estimate can be made from any homogenous region in the image. The assumption is, as long as all pixels contain the same underlying value (due to the reflectance of the material) then a pixel shift of the area should expose a difference in the pixels that is related to noise and not true signal. Though this method is not completely accurate, it does allow for a rough estimate of the sensor noise.

For the exposed imagery, a noise estimate was obtained using a 16 $\times$ 14 pixel calibration panel located in the scene. A possible drawback with using this panel is the number of pixels on it (*i.e.,* a sample size of 224) and the fact that the panel is transparent (*i.e.,* not homogenous). The non-opaqueness of the calibration panel leads to variation in the pixels mainly due to the varying grass structure underneath. Therefore, a second estimate was obtained from a homogenous water region in a flight line that was captured on a different day. This area can be seen in Figure 4.31. Here the sample size is fairly large (*i.e.,* over 1000 pixels) and the area is homogenous. However, it was captured on a different day (and altitude) than the imagery tested in this research. Assuming the sensor noise variation did not change much from day to day, there is still a potential issue of integration time due to different altitudes. That is, the exposed image was captured at 5,000 feet with an integration time of 5 ms while the "water" image was captured at 20,000 feet with an integration time of 15 ms. Previous studies [25], comparing two different

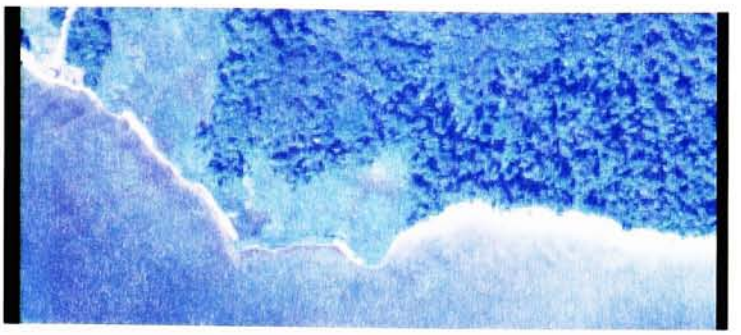
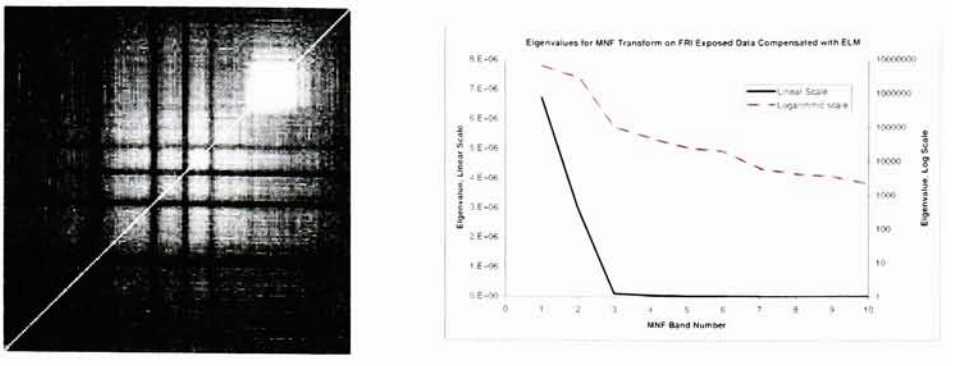


Figure 4.31: HYDICE image used to obtain sensor noise estimate. A 1000 sample region over water, located in the lower left of the image, was used to as input to ENVI's MNF procedure.

integration times both with low signal levels, showed that the HYDICE instrument was fixed noise limited. That is, at these signal levels, the instrument noise level is not signal dependent.

In short, both noise sources were used with the MTMF algorithm and produced similar infeasibility images. Therefore, we proceed in this section by illustrating results (for both the exposed and concealed data sets) based on noise obtained from the "water" image. The noise correlation matrix obtained from this region can be seen in Figure 4.32 along with an eigenvalue plot of the first 10 MNF bands, as applied to the ELM'ed exposed imagery. A similar plot was found when the MNF transform was applied to exposed data compensated with ATREM and FLAASH. Lastly, this trend in variances was also seen when the MNF transform was applied to the concealed ELM'ed imagery. Therefore, from the distribution of eigenvalues, the first 8 MNF bands were selected for MTMF processing.

In summary, we outline the overall process for obtaining the INF images, including estimation of sensor noise and MNF transformation, using the "water image". This outline is as follows:



(a) (b)

Figure 4.32: (a) Shows the 2 dimensional correlation matrix of the noise statistics file associated with the water region while (b) illustrates the sharp fall off in variance associated with the first 10 MNF bands, as applied to the ELM'ed exposed imagery.

1. Perform (temporary) MNF transform on reflectance image used for noise estimate. Select homogenous region over water (or calibration panel, as previously explained). Obtain noise statistics file.

2. Perform MNF transform on reflectance image used in target detection analysis. Use noise statistics file from previous water region image.

3. Obtain reflectance spectrum of target. Resample. Transform spectrum using same MNF rotation as that used on reflectance image containing targets. Convert to an ENVI spectral library file.

4. Analyze the eigenvalue plot to determine how many MNF bands to keep.

5. Run MTMF with inputs from MNF (band reduce) image (containing targets) and converted reflectance spectrum (which was converted to MNF space).

6. Export individual INF image file using "create new file" option in ENVI.

4.4.2 Infeasibility (INF) Images

In this section we visually analyze the infeasibility images that are used in conjunction with the MTMF and PB-SIFT algorithms for both the exposed and fully concealed data sets.

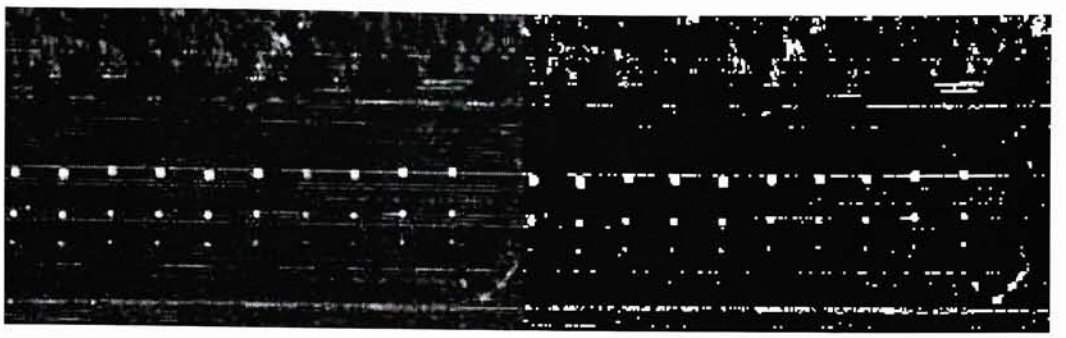
For the exposed imagery, the MTMF algorithm is applied to reflectance imagery (ELM, ATREM, and FLAASH) that has been Maximum Noise Fraction (MNF) transformed. We show results where all the MNF bands are kept as well as only using the first 8 bands. The PB-SIFT algorithm is applied to sensor-reaching radiance data, where all bands were used. Here, we illustrate results from using the mean target space vector.

For the concealed imagery, the MTMF algorithm is applied to ELM'ed reflectance imagery only. This is because of the unavailability of the data, previously mentioned. Here, we show results where all the MNF bands were used in addition to only using the first eight. Similarly, the PB-SIFT algorithm (using the mean target space vector) is applied to the radiance cube, where all bands were used in the analysis.

For each result, exposed and concealed, we show the raw byte scaled version of the INF or SIP result followed by a version that has a threshold applied, illustrating the top scores.

INF - Exposed

The first INF image obtained, was that from ENVI's MTMF using the exposed data set compensated with ELM. As a starting point, all MNF bands were kept in the analysis. This result can be seen in Figure 4.33. In general, this algorithm produces brighter pixels when infeasibility values are large. We can see the byte scaled INF map in Figure 4.33(a). A version of this where only the top 1500 scores



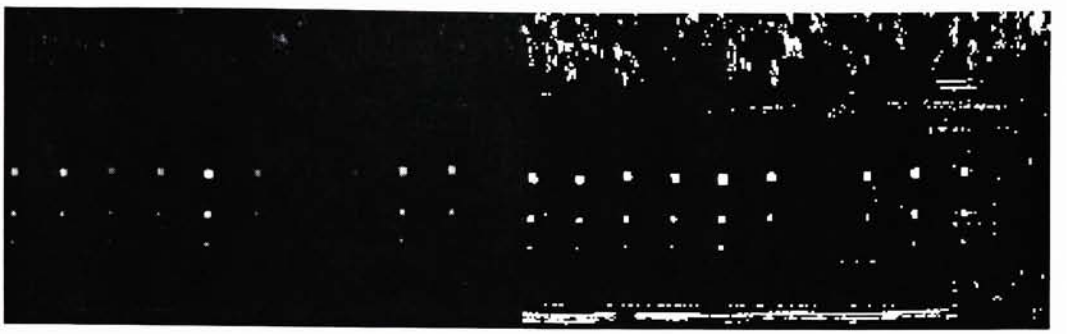
(a) (b)

Figure 4.33: Infeasibility results from ENVI's MTMF algorithm as applied to the FR I exposed data set compensated with ELM. Here, we have keep all MNF bands. The raw test statistic image (a) can be seen next to a version (b) with a threshold applied, showing the first 1500 scores. For the INF algorithm high values (bright pixels) are equivalent to large infeasibility values.

are displayed can be seen in Figure 4.33(b). To generate this image a threshold of 16.1 was applied.

At first glance, this result seems counter intuitive since the algorithm has marked all the targets with large infeasibility scores. Clearly, the target F3 is not infeasible since it is what we are looking for. In the ENVI software, this INF result is computed at the same time as the SMF. We have already shown that the SMF component of the MTMF worked very well in finding the F3 targets in reflectance space. Even though we are now operating in MNF space, we should still obtain the same "detection" results, as long as we keep all the MNF bands. This is possible because the reflectance spectrum is simply transformed using the transformation that was applied to the original data. The only difference is that the INF metric requires a noise covariance estimate which may be the source of ambiguity.

From looking at Figure 4.33, we also notice artifacts or horizontal stratifications. This is most likely due to sensor artifacts. To remedy this situation, we used the

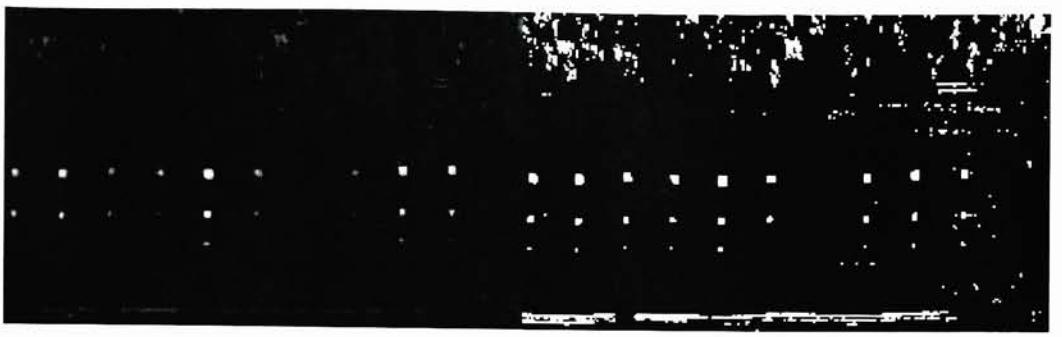


(a) (b)

Figure 4.34: Infeasibility results from ENVI's MTMF algorithm as applied to the FR I exposed data set compensated with ELM. Here, we used the first 8 MNF bands. The raw test statistic image (a) can be seen next to a version (b) with a threshold applied, showing the first 1500 scores. For the INF algorithm high values (bright pixels) are equivalent to large infeasibility values.

MTMF algorithm as it was intended. That is, we analyze the eigenvalue plot and determine a reasonable number of MNF bands to use in the analysis, as was previously discussed. The results of using the ELM imagery and the first 8 MNF bands can be seen in Figure 4.34. We immediately notice that most of the banding or streaking has been removed. However, we still see a number of target pixels (including F3) with large infeasibility scores.

To complete our analysis on the exposed data set, we extracted the INF images from the MTMF algorithm as it was applied to ATREM and FLAASH compensated imagery. We also kept the first 8 MNF bands so as to mitigate some of the streaking artifacts. These results can be seen in Figures 4.35 and 4.36. These results look extremely similar to the result obtained using ELM. We already know, from previous analysis, that there are differences in detection results using different detectors on data that was compensated differently. However, here we do not see a difference in this "combination" of testing. This can only tell us that variations in the INF



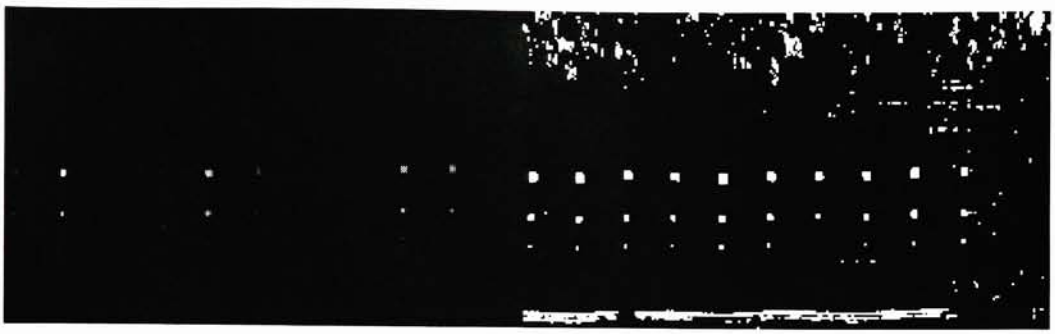
(a) (b)

Figure 4.35: Infeasibility results from ENVI's MTMF algorithm as applied to the FR I exposed data set compensated with ATREM. Here, we used the first 8 MNF bands. The raw test statistic image (a) can be seen next to a version (b) with a threshold applied, showing the first 1500 scores. For the INF algorithm high values (bright pixels) are equivalent to large infeasibility values.

images are more than likely due to noise estimation errors.

Perhaps both estimates of noise (*i.e.*, calibration panel and water image) are more incorrect than previously thought. The rectification of this problem would entail a closer look at the impact noise estimation has on the INF images. Due to the lack of data, this will have to be the subject of future work. Therefore, we proceed with the noise estimates previously obtained and draw careful conclusions knowing this fact.

We then applied the SIP algorithm to sensor-reaching (fully exposed) radiance data, where the orthogonal projection operator was described using the mean target space vector. The results of this can be seen in Figure 4.37. This infeasibility map [*cf.* Figure 4.37(a)] looks completely different to the one generated using the INF algorithm. We first notice that targets F3 and F5 have noticeably low infeasibility values (dark pixels). This is good since we are looking for them, (at least F3), and do not want them to be considered infeasible. It is also noticed that target F2 has a

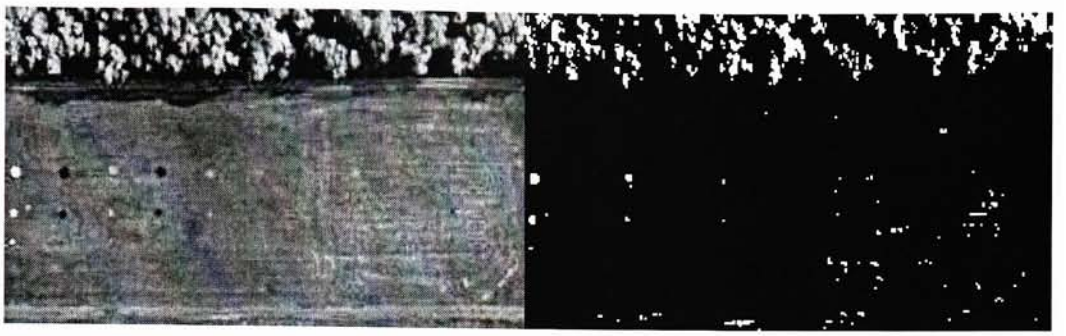


(a) (b)

Figure 4.36: Infeasibility results from ENVI's MTMF algorithm as applied to the FR I exposed data set compensated with FLAASH. Here, we used the first 8 MNF bands. The raw test statistic image (a) can be seen next to a version (b) with a threshold applied, showing the first 1500 scores. For the INF algorithm high values (bright pixels) are equivalent to large infeasibility values.

rather large infeasibility score (bright pixel in the infeasibility map). From the visual imagery of Figure 3.11, we notice that F2 is significantly brighter than anything else in the scene. These results are encouraging and will further separate targets from anomalous and/or potentially bright pixels in our 2 dimension decision space, to be described in the next section.

A version of the byte scaled infeasibility map with a threshold (0.45) applied, can be seen in Figure 4.37(b), which displays the top 1500 infeasibility scores. From this we noticeably see that targets F2 and F4 are considered infeasible along with targets F11, F14, T1 and T2. Many regions in the forest are considered infeasible as well. These results are very encouraging and show that un-target-like behavior can be readily exploited.



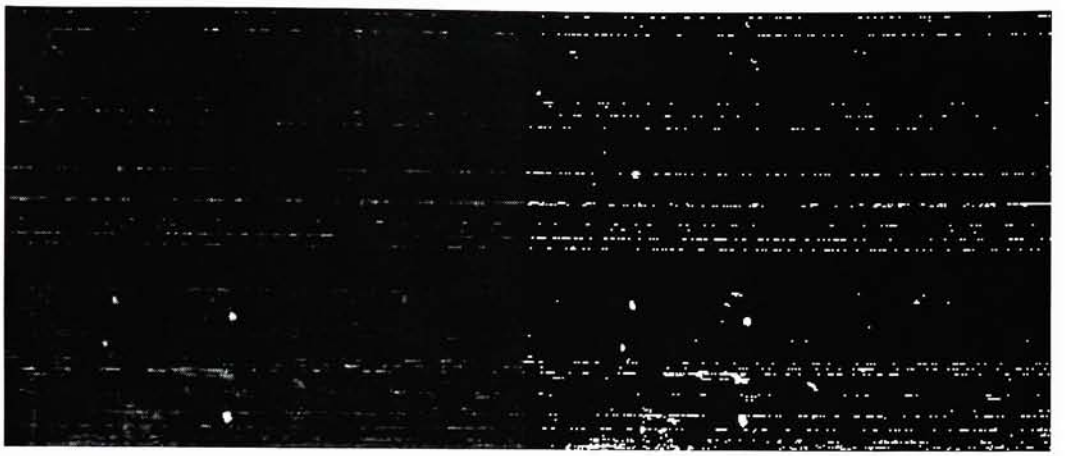
(a) (b)

Figure 4.37: Infeasibility results from applying the SIP algorithm to the FR I exposed radiance data set. The raw test statistic image (a) can be seen next to a version with a threshold applied (b). For the SIP algorithm high values (bright pixels) are equivalent to large infeasibility values.

INF - Concealed

A similar evaluation, to that of the exposed imagery, was performed on the fully concealed test data set. When the INF algorithm is applied to data that has been compensated using ELM, keeping all MNF bands, we get the result seen in Figure 4.38. The infeasibility map of Figure 4.38(a) shows a large score for a point in the forested area believed to be target panel T2 (bright pixel in lower middle of image). Another region in the forested area that shows a large INF score is believe to be related to target panel F14 (not as bright a pixel to the bottom left edge of the image). We also see a third significant INF score associated with a bright, unknown, object in the bottom center of the image. The rest of the background seems to generally have the same INF score overall.

A version of this with a threshold of 20.1 applied can be scene in Figure 4.38(b). Again there is the stratifications due to the MNF transform and sensor artifacting. Since the fully concealed forested scene is so "busy", it is difficult to tell what other



(a) (b)

Figure 4.38: Infeasibility results from ENVI's MTMF algorithm as applied to the FR I concealed data set compensated with ELM. Here, we have keep all MNF bands. The raw test statistic image (a) can be seen next to a version (b) with a threshold applied, showing the first 1500 scores. For the INF algorithm high values (bright pixels) are equivalent to large infeasibility values.

target panels, if any, have been mark as infeasible.

To mitigate against noise variations in the later MNF bands, the INF algorithms was again applied to the ELM compensated imagery using only the first 8 MNF bands. This result can be seen in Figure 4.39. Again, we immediately notice the disappearance of streaking in the imagery. We also notice that from Figure 4.39(a) that some of the pixels marked as highly infeasible, look similar to those previously seen in Figure 4.38(a). The actual ranking of these infeasibility scores is difficult to assess in this representation and will be explored in subsequent sections where more quantitative analysis, using 2 dimensional scatter plots, is performed.

Finally, the SIP algorithm was applied to the sensor-reaching (fully concealed) radiance imagery where the results are shown in Figure 4.40. The infeasibility map

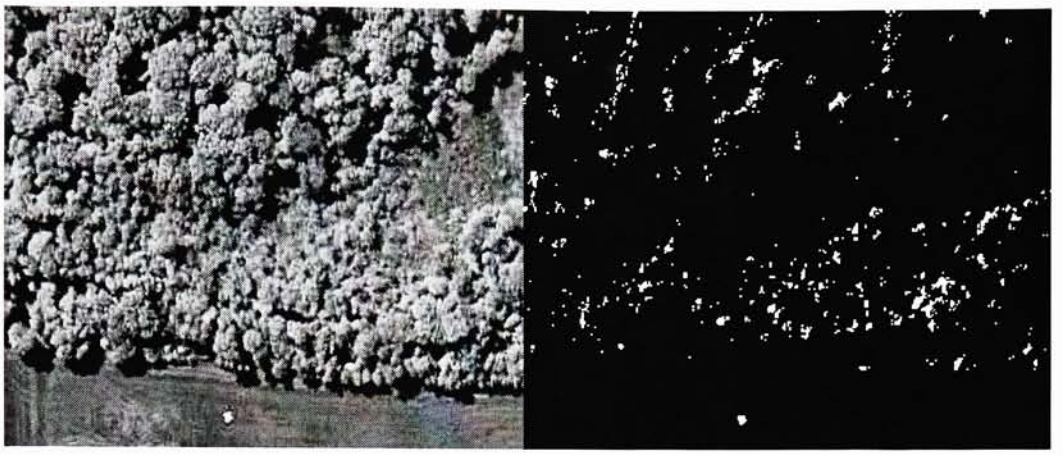
(a) (b)

Figure 4.39: Infeasibility results from ENVI's MTMF algorithm as applied to the FR I concealed data set compensated with ELM. Here, we used the first 8 MNF bands. The raw test statistic image (a) can be seen next to a version (b) with a threshold applied, showing the first 1500 scores. For the INF algorithm high values (bright pixels) are equivalent to large infeasibility values.

of Figure 4.40(a) shows an overall increase in contrast compared to when the INF algorithm was applied. This will be explored more in the next section. From the infeasibility map, we also get a large score, similar to the INF algorithm, associated with a bright, unknown, object in the bottom center of the image. Other significant infeasibility scores are difficult to locate due to the "busy" nature of the scene.

We can get a closer look at the top 1500 scores by applying a threshold of 0.523, as can be seen in Figure 4.40(b). The first thing that is noticed is the lack of MNF/sensor artifacting. Most of the large infeasibility scores are associated with the forested background, with the exception of the unknown, bright object in the grass field.

In summary, the infeasibility images generated using the SIP algorithm, as ap-



(a) (b)

Figure 4.40: Infeasibility results from applying the SIP algorithm to the FR I concealed data set. The raw test statistic image (a) can be seen next to a version with a threshold applied (b). For the SIP algorithm high values (bright pixels) are equivalent to large infeasibility values.

plied to exposed and fully concealed imagery, show promising results. For the exposed imagery, the SIP algorithm identified appropriate pixels as being infeasible while simultaneously scoring those that were target with low infeasibility values. The same can be said for the fully concealed results, though it was not entirely obvious how all the target pixels scored.

The infeasibility scores generated using the MTMF INF algorithm on the exposed data produced counter intuitive results. Target panels were scored with high infeasibility values. This trend was still evident, though to a lesser extent, when a handful of MNF bands were kept in order to circumvent the appearance of streaking due to sensor artifacts. It is unclear, at this time, as to why this is happening. From the above analysis, estimation in sensor noise maybe the culprit.

The infeasibility scores generated using the MTMF INF algorithm, as applied to

the concealed imagery looked reasonable, though it was difficult to tell if all of them were associated with target panels. The application of fewer MNF bands reduced sensor artifacts, as expected.

4.4.3 2 Dimensional Decision Spaces

This next section puts together the results from the infeasibility algorithms and detection algorithms and displays them as a 2 dimensional decision spaces. We compare results from ENVI's MTMF with the PB-SIFT algorithm, for both the exposed and fully concealed data sets. For the PB-SIFT algorithm, we show results using 3 different permutations of the target space. The first result involves using the entire target space or all 180 target space vectors in the projection operators. The second result uses only the 7 basis vectors found with MaxD, while the third result populates the projection operators with the target space mean vector.

PB-SIFT and MTMF - Exposed

The application of using a detector with an added infeasibility metric, as applied to the FR I exposed imagery in radiance space, is summarized in the 2 dimensional plots of Figure 4.41. Figure 4.41(a) shows the result of using the PB-SIFT algorithm where the projection operators are described using all the target space vectors. We first notice that the target pixels have reasonably large separation from the background distribution (in gray). This is similar to results using just the PBosp detector. We can see that the background (in all figures) has a low abundance while the target pixels have abundances ranging from 0.2 to almost 1.0. If we were to threshold this space on the y-axis (PBosp) only, say at an abundance of 0.2, which is just above the background, we would take on a number of false alarms related to panels F5, F11,

T1, and T2. However, we can mitigate some of these false alarms by considering the infeasibility scores from the SIP. Looking at the SIP axis we see that the background has a range of infeasibility values associated with it, with some of them being more feasible than actual F3 target pixels (*i.e*, shorter distance to the target space). More importantly is the spread in distances among the various panels. We see that panel F11 has almost the same projection as F3 onto the PBosp axis but much different projection onto the SIP axis. Separation along the SIP axis must exist in order to increase detects using the PBosp detector. Looking at this plot clearly shows that we can not use the SIP metric alone in detection, for the target pixels would be lost in the background. This is true of all the plots in Figure 4.41.

When populating the projection operators with 7 basis vectors we see an improvement in the decision space, as can be seen in Figure 4.41(b). The improvement is by way of the PBosp detector and not so much the SIP metric. However, it can be seen for the first time that one might be able to separate target F3 from F5, though further evaluation would need to be performed for confirmation. This potential separation would not be possible using the PBosp detector alone.

When the mean target space vector is used, as in Figure 4.41(c) we see a significant separation among panels and background. The improvement is seen in both the PBosp detector as well as the SIP operator. Looking at the PBosp axis we see more separation between targets F3/F5 and the background, though now we can't separate F3 from F5. Furthermore, there is a greater separation between F3/F5 and the rest of the panels. However, using PBosp alone will still result in false alarms due to panels T1, T2, and maybe Fll and F5 (though we could just consider F5 the same "target" as F3). Therefore we turn to the SIP axis. Here we see overall improved separation between vector distances, though the separation between

F3 and F5 has now decreased. The F11 pixels have increased their distance from the F3 target pixels. We additionally see an overall increase in the spread of the background distribution.

If the ideal decision boundary was established, we could probably obtain all the F3 and F5 target pixels (if we treated F3 and F5 as being the same target) no matter how many target space vectors we used in the projection operators. This is because the separation between the target group (F3/F5) and the background/other panels, is about the same in all three cases presented.

We now turn our attention to the MTMF algorithm, as implemented using ENVI software. This algorithm combines the SMF and INF algorithms previously mentioned. This algorithm was run on MNF transformed reflectance imagery that originated from using compensation algorithms such as ELM, ATREM, and FLAASH. These 2 dimensional decision spaces can be seen in Figure 4.42. We start by looking at the decision space created using ELM compensated imagery. For comparison purposes, we generate a decision space keeping all (170) MNF bands, as can be seen in Figure 4.42(a). Initially, the results from the SMF (y-axis) are in-line with those we found previously (*cf.* Sec. 4.3). That is, there is reasonable separation of the target pixels F3 and the background. In addition, the remaining target panels are fairly separated from the background as well. However, the infeasibility scores for the target pixels F3/F5 seem much too large. That is, the INF metric has marked most of the target pixels more infeasible than the background, which seems counter intuitive. The actual cause of this is still unknown but is more than likely due to noise estimation issues, as previously discussed. A decision space, such as the one in Figure 4.42(a), would make it very difficult to generate an automated decision boundary that separates target pixels from backgrounds and other panels. However,

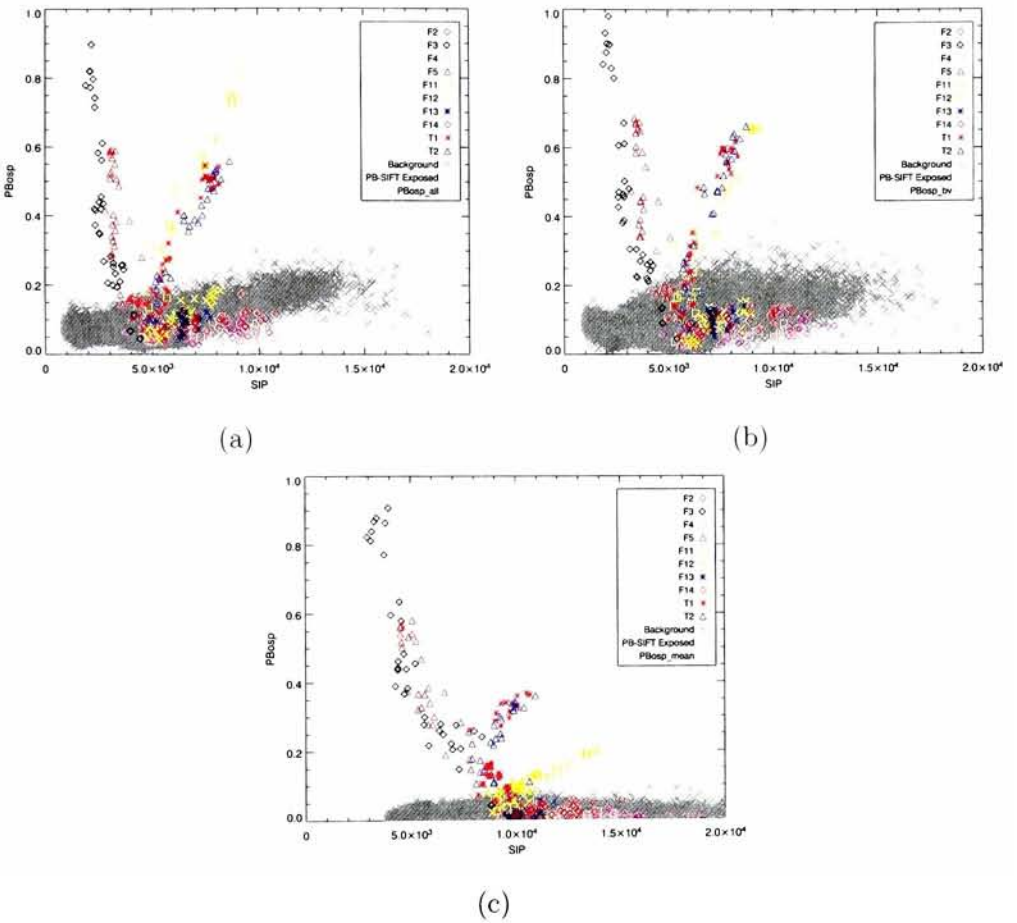


Figure 4.41: Two dimensional decision spaces created using the PB-SIFT algorithm, as applied to the FR I exposed data set. (a) Shows the result of using PB-SIFT, using the entire target space, (b) shows the result of using PB-SIFT, using 7 basis vectors, and (c) shows the result of using PB-SIFT, using the mean target space vector.

a user could manually select this region with moderate success by identifying about a third of the F3 pixels with out any false alarms.

The MTMF was then applied to (exposed) MNF transformed data where only the (eight) most relevant variance bands were kept, which is how the algorithm is typically used. Applying this to the ELM compensated imagery results in the decision space of Figure 4.42(b). Here, we see a much larger separation between all target signatures and background. Furthermore, we see a distinct separation among the targets themselves, particularly between F3/F5 and the rest of the target panels. As was seen in Figure 4.41(b), we now notice some separability between targets F3 and F5. However, we still notice that the targets F3/F5 have large infeasibility scores, relative to the background. Though this is not as pronounced as was seen in Figure 4.42(a), it still leads us to believe that poor sensor noise estimation is causing this phenomenon. In general, fewer number of MNF bands improved the decision space.

Similarly, the MTMF was applied to MNF transformed, FR I exposed data that was compensated using ATREM and FLAASH. These results can be seen in Figures 4.41(c) and 4.41(d). Again, only the first eight MNF bands were used in the analysis. Figure 4.41(c) shows a general improvement in the SMF algorithm as compared to the ROC curve of Figure 4.23. That is, the SMF performed poorly on the raw reflectance imagery. However, when moving to a reduce MNF space, we see that the SMF can now isolate about one third of the F3 pixels with out any false alarms, which was previously not evident.

The decision spaces produced by using ATREM and FLAASH compensated imagery are about the same, though the ATREM result is slightly better. However, neither is as good as the decision space obtained with ELM compensated imagery,

as expected.

PB-SIFT and MTMF - Concealed

In a similar manor to that of the exposed data set, the PB-SIFT algorithm was applied to the FR I fully concealed imagery. These decision spaces can be seen in Figure 4.43. Once again, three variations of the target space are explored. One variation using all the target space vectors to describe the projection operators, while the other two consist of using 7 basis vectors and the mean target space vector.

The results of using all the target space vectors for the projection operators can be seen in Figure 4.43(a). In this result the abundance of the background is very high and essentially masks out the target pixels making them nearly impossible to detect, even with the added SIP metric. This is a clear reason why not to use all the target space vectors in the projection operator. There is simply too much overlap with the background.

When using the 7 endmembers previously discussed, we get the 2 dimensional decision space of Figure 4.43(b). The abundance of the target pixels has stayed the same while the overall abundance of the background has diminished significantly. If we were to use the PBosp detector alone, while incrementally lowering the threshold, we would incur false alarms on F14, T2, and then background. Using the SIP metric here with the PBosp detector would reduce false alarms significantly, since all the targets have a similar (small) projections onto the target space.

The overall separation from the background is further increased if we use the mean target space vector, as can be seen in Figure 4.43(c). All the the panels, with the exception of F14, are clustered together with small SIP values. In all PB-SIFT cases, the use of the PBosp detector would certainly result in false alarming on F14.

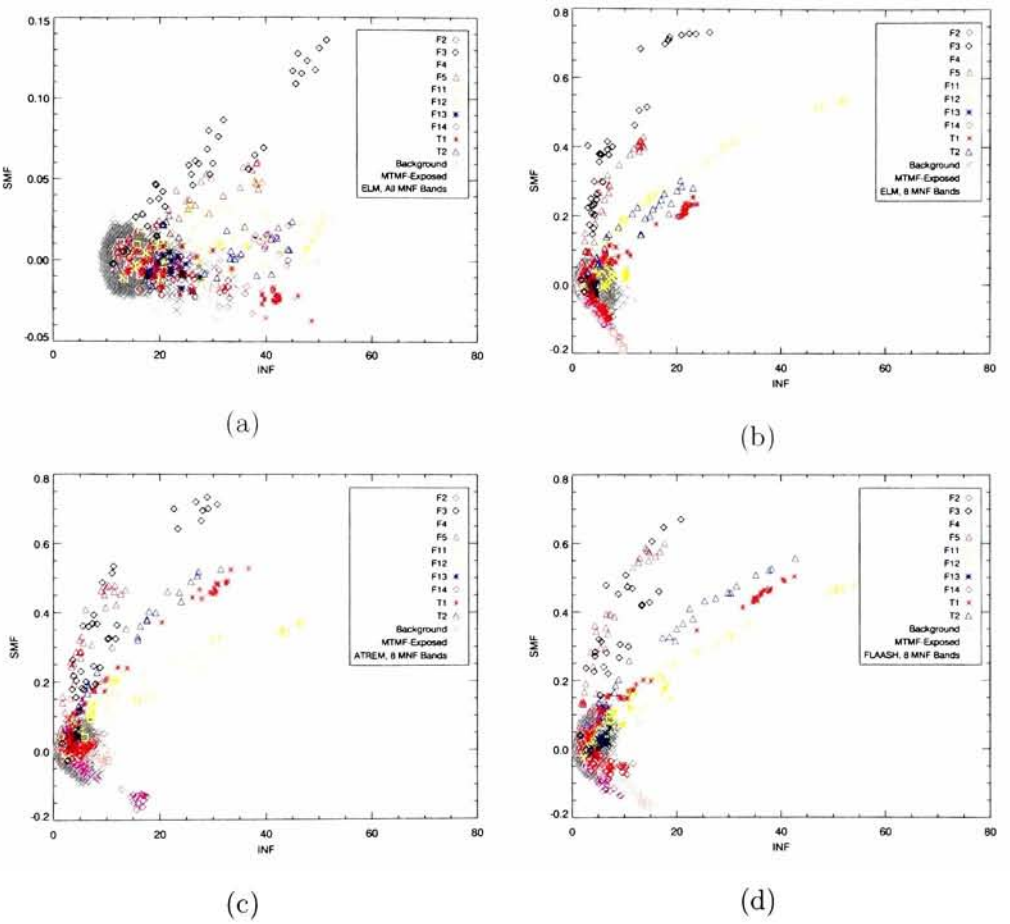


Figure 4.42: Two dimensional decision spaces created using the MTMF algorithm, as applied to the FR I exposed data set. (a) Shows the result using ELM data with all (170) MNF kept while (b) only keeps the first eight. Decision spaces are also shown using (c) ATREM and (d) FLAASH compensated imagery.

However, the utilization of the SIP metric makes it very easy to mitigate this high abundance, high infeasibility, panel.

Finally, we apply ENVI's MTMF to the FR I fully concealed imagery. Unlike the previous analysis, we only tested the algorithm on ELM compensated imagery, which can be seen in Figure 4.44. For comparison, the MTMF was applied to MNF transformed imagery where all the MNF bands were kept [*cf.* Figure 4.44(a)]. This is a more intuitive result than was seen, previously, using the exposed imagery. That is, we notice that the the targets stick out above the background distribution and have relatively low infeasibility scores. In this image we could threshold using the SMF alone and get better results than using PBosp. The SMF would obtain about 50% of the target pixels before the first false alarm while the PBosp detector (using any number of target space vectors) would false alarm after the first target pixel. However, because the background is so tightly clustered "under" the targets (*i.e.*, much of the background has a similar infeasibility score), the utilization of the infeasibility provides no added benefit, for this specific image case. It is seen, however, that the INF metric has assigned larger infeasibility scores to panel F14, which is in-line with the results observed using the SIP metric.

Finally, the MTMF was applied to (ELM'ed) MNF imagery where only the firsts eight bands were kept. This result can be seen ing Figure 4.44(b). Unlike the previous case, all the target panels are buried in the background space. This result seems surprising since previously operating in a reduce space increase general performance. In this case, there is no way to separate targets from background. One possible explanation for this is that the targets have a very low signal to noise, such that when we truncate in the MNF space, we also are truncating valuable target information that would be normally regarded as variation due to noise.

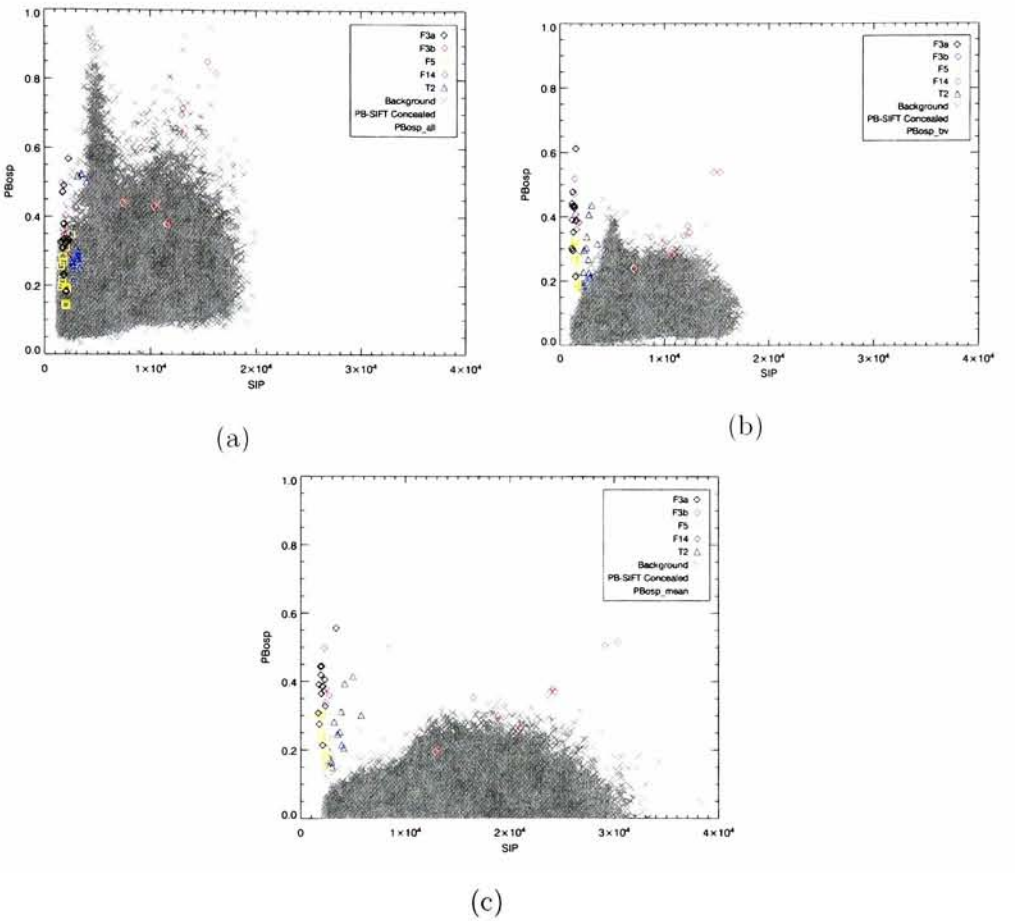
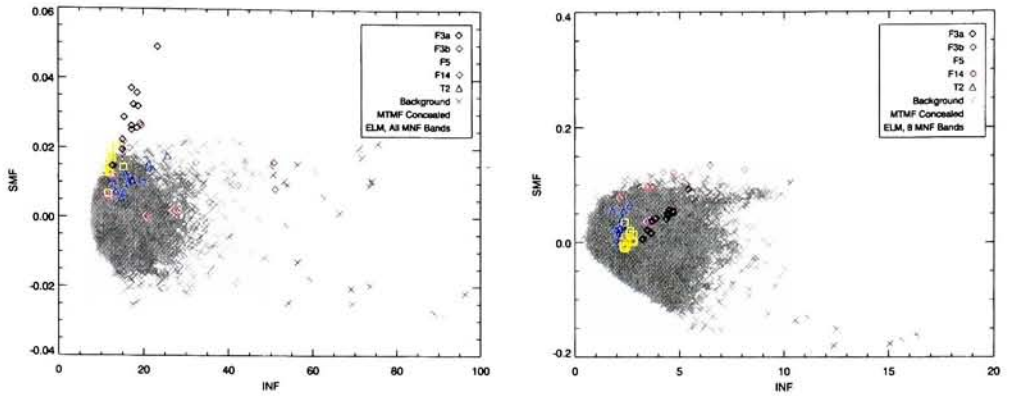


Figure 4.43: Two dimensional decision spaces created using the PB-SIFT algorithm, as applied to the FR I concealed data set. (a) Shows the result of using PB-SIFT, using the entire target space, (b) shows the result of using PB-SIFT, using 7 basis vectors, and (c) shows the result of using PB-SIFT, using the mean target space vector.



(a) (b)

Figure 4.44: Two dimensional decision spaces created using the MTMF algorithm, as applied to the FR I concealed data set. (a) Shows the result using ELM data with all (170) MNF kept while (b) only keeps the first eight.

4.4.4 Improved Performance Varying Threshold of SIP

This section looks at the impact a simple threshold in the SIP value can have on overall detector performance. We apply the SAM and SMF detectors on ELM compensated imagery while the PBosp algorithm is applied to radiance data, as before. This time, however, we use a truth mask that allows us to false alarm on everything in the scene, including the other man-made targets. This ROC curve result is shown in Figure 4.45(a). These curves are similar in nature to those seen in Figure 4.22(a) with the exception that there is now more confusion or false alarms from the other panels.

The idea for improvement is to compute new ROC curves for various SIP thresholds. This was done using results from the "PBosp_bv" case of Figure 4.41(b) where we applied the PBosp detector to imagery using 7 target space vectors. From Figure 4.41(b), we can clearly see that a simple linear threshold will remove false alarms

from almost all the other man-made panels, in addition to mitigating most of the background. This application of a shifting SIP threshold can be seen in Figure 4.45(b). We have now increased the performance of this detector such that it outperforms almost all our previous results, as applied to the exposed imagery. Even with a simple linear threshold, this is an encouraging result.

Using this type of threshold, there exist cases where target pixel SIP scores are actually *larger* than the given SIP threshold. In this case, the targets are intentionally misclassified as background. Two cases of this can be seen in the ROC curve of Figure 4.45(b) as (nearly) vertical lines close to the detection rate axis. Furthermore, we notice that these curves do not have a detection rate approaching one. This is because, in the detection rate calculation, we now have to exclude those targets pixels with SIP values greater than the given threshold while still dividing by the original total number of targets. This produces a maximum detection value less than one. A summary of the thresholds used can be seen in Table 4.4 along with detection and false alarm rates.

We can analyze this table in a number of ways. If we wish to find all 35 F3 targets using the PBosp detector alone, we will false alarm on 73% of the pixels. However, when we introduce the SIP threshold, still seeking all 35 pixels, we now only false alarm on 43% of the pixels. Furthermore, in using a SIP threshold, we can see a trade off in detection versus false alarms. That is, we can reduce our false alarms to zero, however, at a price of finding a maximum of 60% of the targets. This is because the remaining 40% of targets have SIP values greater than the applied threshold and are classified as background. The only way to obtain zero false alarms using the PBosp detector alone (*i.e.,* no added SIP metric), is to capture a maximum of 23% of the targets (8 out of 35 panels) [*cf.* Figure 4.41(b)], a much lower number

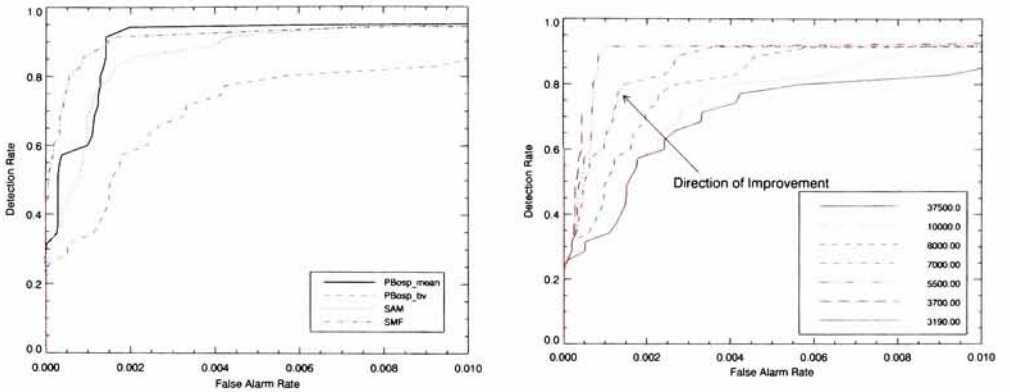
Table 4.4: Summary of SIP thresholds as applied to a 2D decision space generated from the FR1 exposed imagery. Table contains number of pixels with equivalent *-rates* in parenthesis.

Total Target Pixels	False Alarms	Threshold
35 (1.00)	27580 (0.73)	37500 (1.00)
35 (1.00)	26301 (0.70)	10000 (0.26)
35 (1.00)	25219 (0.67)	8000 (0.21)
35 (1.00)	24344 (0.64)	7000 (0.18)
35 (1.00)	15984 (0.43)	5500 (0.14)
25 (0.71)	16 (4.2e-04)	3700 (0.10)
21 (0.60)	0 (0.0)	3190 (0.08)

than was achieved using the SIP metric.

Finally, we apply the this same analysis to the FR I concealed imagery. Again, we generate ROC curves using a truth mask that allows us to false alarm on every thing in the scene, including the other target panels. This result can be seen in Figure 4.46(a) where the SAM result is, again, at the bottom of the plot. These results are comparable to those scene in Figure 4.30(a) with the exception of a scale change in the FA axis. By shifting the SIP value in Figure 4.43(b), we can almost find every F3 target panel with zero false alarms. This dramatically improved result for the PB_osp can be seen in Figure 4.46(b). A summary of the thresholds used can be seen in Table 4.5 along with detection and false alarms rates.

We can perform a similar analysis of Table 4.5 to that obtained when using the exposed imagery. If we wish to find all 18 F3 targets using the PBosp detector alone, we will false alarm on 6% of the pixels (or 4758 image pixels). However, when we introduce the SIP threshold, still seeking all 18 pixels, we now only false alarm on



(a) (b)

Figure 4.45: ROC curve improvement using various SIP thresholds, as applied to the FR I exposed imagery. The SAM and SMF algorithms were applied to ELM compensated imagery. (a) Shows ROC curves using a truth mask that allows for FA's on other man-made panels while (b) shows an increased ROC performance for the PBosp_bv case.

7 pixels, which is about 0.009% of the total image pixels. Additionally, using the PBosp_bv detector alone, we can achieve a zero false alarm level with a penalty of capturing a maximum of 22% of the targets pixels (4 out of 18 targets). However, with the added SIP metric, the same zero false alarm level can be achieve with a reduced penalty of capturing a maximum of 39% of the targets (7 out of 18 targets). That is, we found an additional 3 targets using the SIP metric.

In summary, this section introduced how a simple linear SIP threshold can dramatically improve results for the PB-SIFT algorithm. However, this same analysis could not be performed using the MTMF because of the infeasibility scoring of target pixels. That is, in most cases a varying INF threshold did not provide much benefit in the detection scheme. These MTMF decision spaces need to be created with known sensor noise sources before any definitive conclusions can be drawn.

Table 4.5: Summary of SIP thresholds as applied to a 2D decision space generated from the FR1 concealed imagery. Table contains number of pixels with equivalent *-rates* in parenthesis.

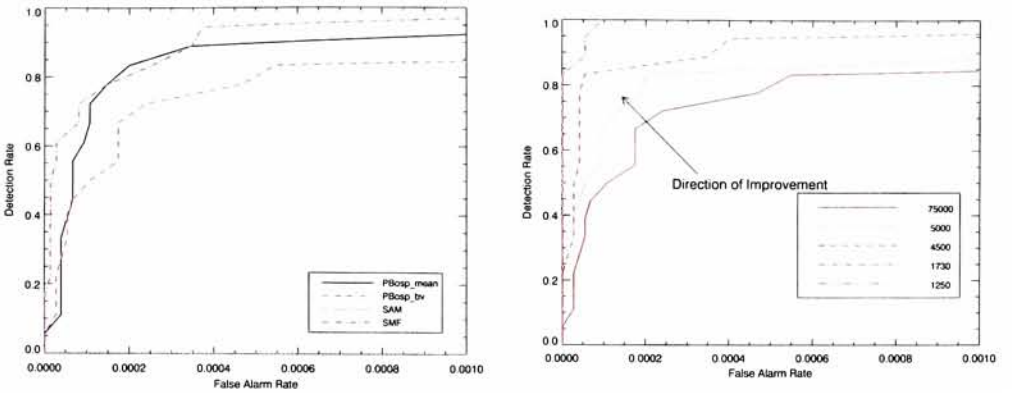
Total Target Pixels	False Alarms	Threshold
18 (1.00)	4758 (0.063)	75000 (1.00)
18 (1.00)	910 (0.012)	5000 (0.066)
18 (1.00)	202 (0.003)	4500 (0.060)
18 (1.00)	7 (9.3e-05)	1730 (0.023)
7 (0.39)	0 (0.00)	1250 (0.167)

4.5 Pixel Behavior in 2D Decision Space

In this last section we first examine the (expected) trend exhibited in the 2 dimensional decision spaces. That is, we implement an image wide linear mixing scheme that shows what the behavior of the PBosp/SIP values are as a function of target-pixel abundance. This is followed by the potential impact a bright or saturated pixel can have on the detection process. More specifically, we introduce saturated pixels in the imagery while examining their behavior in the 2 dimensional decision space. We then illustrate how one can mitigate against such phenomena.

4.5.1 Target Pixel Behavior

The previous section illustrated the advantage of combining a detector with an added infeasibility metric. Specifically, we showed that the PBosp detector combined with the SIP operator, produces 2 dimension decision spaces that enable the user to identify areas where the most probably targets will occur. It is seen in almost all the plots of Figure 4.41, that the ideal decision boundary in the SIP axis should



(a) (b)

Figure 4.46: ROC curve improvement using various SIP thresholds, as applied to the FR I concealed imagery. The SAM and SMF algorithms were applied to ELM compensated imagery. (a) Shows ROC curves using a truth mask that allows for FA's on other man-made panels while (b) shows an increased ROC performance for the PBosp_bv case.

not be a linear one. Additionally, we see that the general trend of target abundance versus infeasibility seems to be similar in all the plots. For this *behavior* analysis, the FR I exposed data was used. Additionally, the projection operator was described using the mean target space vector.

The concept of the expected behavior of target pixels in this space was explained in Sec. 3.2.3. In this section we go ahead and test the theory of what the expected trend in the target pixels should be. That is, we look at the SIP mean distance and variance, as a function of pixel abundance.

To test this behavior, we simply use a linear mixing model that adds a known percentage of target to every pixel in the image. We then run the detection using this "mixed" background data. This is then repeated for various target percentages.

The mixing approached used was of the form

$$\mathbf{z}_i = \alpha\,\mathbf{t} + (1-\alpha)\mathbf{x}_i \quad \text{for } i = 0, 1, \ldots, N \tag{4.8}$$

where $\mathbf{z}_i$ is the newly mixed background pixel, α is the fractional amount of target, $\mathbf{t}$ is an actual target pixel from the image, $\mathbf{x}_i$ is an image pixel, and N is the total number of pixels in the image.

The target, $\mathbf{t}$ came from a fully resolved F3 pixel found in the exposed data set. Fractional values tested included 0%, 20%, 40% and 70%. The results of this target-mixed background can be seen in Figure 4.47. As a base-line the 0% fill factor was generated. This can be seen as a cluster of black pixels near the bottom of the graph. This result is exactly the same as what was observed in Figure 4.41(c). When using a fill factor of 20% (magenta pixels) the overall abundance expectedly increases. Additionally, the variance of the distribution decreases. This is because we are slowly "taking out" background variability from the pixels. The mean projected distance also decreases, as expected. This is because the overall cloud of data is converging to the target pixel, $\mathbf{t}$, which has a specific (short) distance to the target space. As we add more target to each background pixel, 40% (red pixels) and 70% (blue pixels), we see the general trend in reduced mean-distance and variance continue. This process continued until the distribution converged to a single point with no variance and a projected mean equal to that of the target pixel $\mathbf{t}$'s projected distance to the target space.

This confirms our intuition about the behavior of the SIP operator in the 2 dimensional decision space. We have already shown that each component of the PB-SIFT algorithm can out perform the SAM, SMF, and MTMF algorithms, though we would should analyze the noise issue associated with MTMF algorithm more closely. This result was obtained without taking into account the expected behavior of the

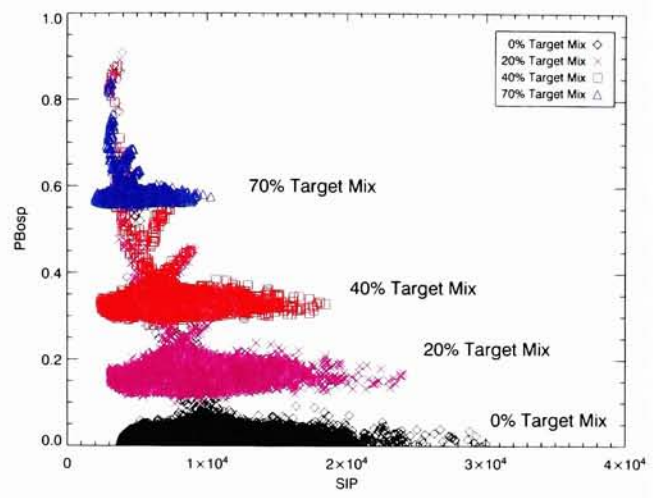


Figure 4.47: Results from testing the expected behavior of the Structured Infeasibility Projector (SIP). We can see that as the target abundance in a background pixel increases, the overall variance and mean projected distance decreases.

decision space. Creating a decision boundary while taking advantage of this new knowledge can only increase performance beyond an already promising approach to detection.

4.5.2 Saturated Pixel Behavior

The PBosp detector used in this research is a type of matched filter which makes it susceptible to leakage or false alarms due to bright or saturated pixels. This type of pixel appears target-like to the detection algorithm but is truly not target. To illustrate this phenomena, a hand full of background pixels from the FR I concealed imagery was used to simulate bright or saturated pixels.

This was accomplish by first locating three background grass pixels in the FR I concealed image of Figure 3.15. These three pixels were then given radiance values

of 8000, 12000, and 13000. The 'saturated' radiance values were only applied to bands 1 and 2 and were derived by examining the maximum radiance values that normally occurred in these bands. Typically, maximum values ranged from 8000 to 9000 micro-flicks in these earlier bands. The detection algorithms (PBosp and SIP metric) were applied to the data set which resulted in the generation of the 2 dimensional decision space shown in Figure 4.48. Here, we used the mean target space vector in the detection process to produce a result similar to that of Figure 4.43(c), with the exception of the implanted saturated pixels.

It is clear that the PBosp detector would false alarms on these pixels had it been used alone, for they are above the background and exhibit large target-like (or abundance) behavior. However, utilizing the SIP metric, we can mitigate against such false alarms by keeping our un-target-like SIP threshold reasonably low. In this case, the saturated pixels are fairly non-target-like and therefore would be excluded in the overall detection process, even when a 'loosely' set SIP threshold was applied.

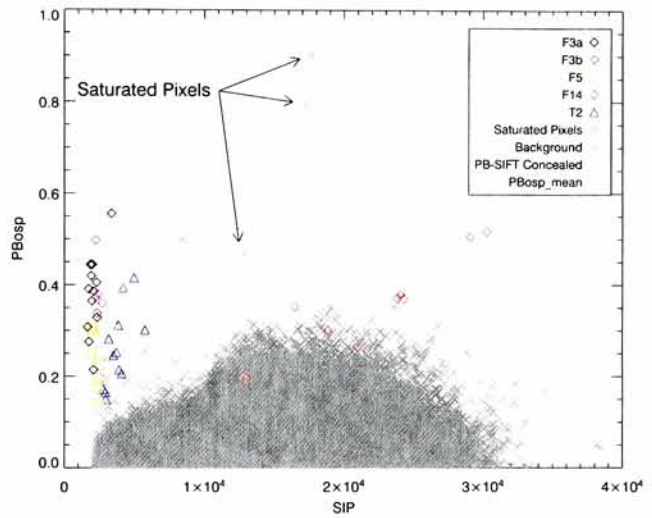


Figure 4.48: Two dimensional decision space created using the PB-SIFT algorithm, as applied to the FR I concealed data set. This example illustrates the impact saturated pixels have on the detection process.

"The science of today is the technology of tomorrow."

Edward Teller

5

Conclusions and Recommendations

5.1 Summary

The research presented in this thesis explored new methods of improving target detection using the concept of physics based modeling. The work builds upon an original body of work related to detection using illumination invariant subspaces. In this research, we, not only reproduced the original work, but refined the process of creating target subspaces. We then developed a detector that could adapt to such target spaces, called the Physics Based Orthogonal Subspace Projection (PBosp) algorithm. This detector produced a quantity synonymous with the abundance of target signature in a test pixel. In addition, we extended this detection process to incorporate an added infeasibility measure. This idea was born from the concept of

the Mixture Tuned Matched Filter (MTMF), which is stochastic in nature. After deriving this algorithm, a geometric equivalent (in the sense that we create 2D decision spaces) was developed. This algorithm is called the Physics Based Structured InFeasibility-detector (PB-SIFT), and uses a newly developed projection operator called the Structured Infeasibility Projector (SIP), which produces an infeasibility measure similar to that found in the MTMF. The expected behavior of the SIP metric was explored, as a function of abundance, so as to help in the generation of 2 dimensional decision boundaries. This method of detection was tested using the Forest Radiance I hyperspectral data set using targets that were fully exposed to the sky and ones that were fully concealed in a forested area. These results were then compared to other detection algorithms, namely the Spectral Angle Mapper (SAM) and Spectral Matched Filter (SMF). It was determined, through ROC curve and visual analysis, that all three algorithms (PBosp, SIP and PB-SIFT), based on the imagery used in this research, out performed SAM, SMF and MTMF. One caveat is the noise issue related to the MTMF, which may have hindered results normally obtained using this algorithm.

5.2 Specific Research Conclusions

The new approach in this research first consisted of improving a previous implementation of a physics based approach to target detection called the illumination invariant method. Here thousands of spectra were created, using MODTRAN, in a method that did not take advantage of some of the physical process and interactions involved when generating target spaces, which is what physics based modeling produces. This research demonstrated how one can use MODTRAN to create spectra to populate such target spaces by only requiring the user to know some common a

priori knowledge about the scene, such as atmospheric model, scene elevation, time of day, etc. The user is also required to know (or estimate) varying parameters such as visibility, sensor elevation, and water vapor scale factor. Since the optimal range and values of these parameters could be challenging to estimate, a framework has been established that shows the user how to select the most appropriate range and values for such parameters. This includes the use of external estimation algorithms such as those used to estimate column water vapor. The reasoning for such selections is also explained in addition to what to do when one does not have information on a particular parameter. These parameters, and their ranges, were used to generate sensor-reaching radiance spectra which were stored in look-up-tables.

The governing sensor-reaching equation from the original implementation of the invariant method was modified to include additional parameters such as target orientation (to account for projected area effects) and shape factor to modulate the amount of downwelled radiance. After implementing such post process operations, target spaces that were 180 vectors in size were created for the Forest Radiance I hyperspectral data set. Two such target spaces were created, one for a 170 band hyperspectral image containing targets in an open field and one for targets that were fully concealed in an adjacent forested area. These target spaces were then evaluated using a simple linear mixing model and showed that they can reproduce a HYDICE image target pixel spectrum to less than 1% RMS error (equivalent reflectance) in the visible and less than 6% in the near IR.

Target spaces were then represented with endmembers using the Maximum Distance method (MaxD). Background endmembers (also computed using MaxD) were computed based on a background/target space augmented data set. The concept here was to augment the background with the target space in hopes of shielding the

actual image target pixel, so it does not become a background endmember. This was performed based on multiple image targets. The target spaces did not always shield the image target pixel and half the time, depending on the number of endmembers one selected, the target pixel would become an endmember. This will be the subject of ongoing work.

Detection algorithms were then applied to hyperspectral FR I data utilizing the physics based target spaces and geometrically described background spaces. The algorithms tested included SAM, SMF and PBosp. The SAM (geometric in design) and SMF (stochastic in design) algorithms were applied to ground leaving reflectance imagery that was atmospherically compensated using the ELM, ATREM, and FLAASH algorithms. The PBosp algorithm was applied to radiance data utilizing physics based target spaces.

Results on the FR I exposed and fully concealed imagery showed that the PBosp detector, using the mean target space vector, out performed both SAM and SMF, (using ELM, ATREM and FLAASH) as well as the PBosp, using 7 basis vectors. One notable difference was that the SAM algorithm performed exceptionally poor when tested using the fully concealed imagery. However, the 7 basis vector case performed as well, or better than the SAM and SMF algorithms when reflectance imagery compensated with ATREM or FLAASH was used. In general, the ATREM and FLAASH results are more in-line with typical detection applications.

We then tested and evaluated ENVI's infeasibility (INF) metric (in conjunction with ELM, ATREM, and FLAASH compensation algorithms) and compared it to the newly developed Structured Infeasibility Projector (SIP). The comparison of infeasibility images, as applied to the exposed and fully concealed data sets, showed that the SIP algorithm, used as an infeasibility measure, looked very promising.

The SIP algorithm identified appropriate pixels as being infeasible while simultaneously scoring those that were target with low infeasibility values. The infeasibility scores generated using the MTMF INF algorithm produced counter intuitive results, when applied to the exposed imagery. This may have been caused by the noise estimation process. Additionally, the INF algorithm produced results that contained "streaking" originating from the MNF transform and ultimately sensor noise. This streaking artifact was removed when eight MNF bands were used in the INF process. Results from the SIP algorithm did not contain any of these artifacts.

Both the PBosp and the SIP algorithms were then combined to form the Physics Based Structured InFeasibility-detector (PB-SIFT). This detector was compared to ENVI's stochastic MTMF which incorporates the previously described SMF and INF algorithms. Two dimensional decision spaces were created based on truth masks of the imagery. Results showed the the PB-SIFT algorithm, using the mean target space vector, generated decision spaces that were as separable (if not more so) than those created using the MTMF approach. This same conclusion was drawn for the fully concealed imagery as well. In general the only thing the PB-SIFT algorithm requires is a basis vector description of at least the background as input. One can also provide it with a basis vector description of the target space as well. The MTMF requires the user to provide it with an MNF transformed image which requires an image noise estimate.

We then analyzed increased performance of the PB-SIFT algorithm by applying a varying SIP threshold. These results showed that a simple linear SIP threshold can dramatically improved detector performance. This analysis was not performed on the MTMF output due to the counter intuitive nature of the decision spaces. That is, visual inspection showed that applying a varying INF threshold would produce

little added benefit to detection results. This was apparent for both the exposed and concealed imagery, which used ELM, ATREM, and FLAASH compensation algorithms. This trend was also observed when all MNF bands, as well as the first eight, were kept. It is believed that the source of this artifact has its roots in the noise estimation process. This is the topic of on going investigations.

Finally, the general behavior of the SIP metric was evaluated. This was accomplished by applying the detector to background data that was mixed with a target signature in a specified amount. Using a linear mixing approach, image pixels were mixed with a certain fraction of target signature. These pixels were then used as background data in the detection process. Various mixing fractions showed that the trend of the background data tends to form an (infeasibility) cone. As a function of target abundance in a given pixel, the background variance was shown to decreases to zero while the mean of the projected distances converged to the mean distance associated with the distance from the target space to the image target.

5.3 Recommendations and Future Work

5.3.1 Physical Modeling and Target Spaces

This research explored the entire chain involved in typical target detection. While most algorithms work on ground leaving reflectance imagery, this research generates target spaces so as to work in a sensor-reaching radiance space. Since this process is relatively unexplored, there is the potential for improvement and continued investigation.

One of the more interesting areas to further investigate is the whole concept of physics based modeling and target space generation. This research has established

the processing chain in developing these spaces in addition to providing some intuitive and concrete advice on how to select parameter values and ranges. What has not been tested is the quantitative impact a change in one parameter has on detection results. This is in-line with a sensitivity study of the various parameters used in the MODTRAN runs as well as the post processing terms such as target orientation and shape factor. In addition, the target space needs to include sensor noise and calibration errors. These parameters also need to be evaluated in terms of their impact on detection.

Another refinement to the target space creation process is to make the target spaces more probabilistic. Currently we get a set of spectra from MODTRAN that reflect changes in the atmosphere, based on user defined input parameters and values. In the detection process, we found that using the mean target space vector sometimes produced the best results. However, is this due to the the mean spectrum really being the closest possible spectrum to the target pixel or the fact that we have simply reduced the number of vectors used in the detection process? That mean target space vector may reflect a good approximation to the actual pixel but certainly this is not the ideal probabilistic space. We intuitively know that the extreme values for visibility, sensor elevation, and WV scale factor, for example, will produce a spectrum that is less likely to be like the target pixel, however, we currently do not have a scheme in place to take advantage of such knowledge. All the input parameters have "equal weighting". If we could change this weighting factor to be Gaussian, for example, then the output spaces would have a mean and variance that would more accurately reflect the importance of the input parameters used in the target space generation process.

One of the key factors in using vector geometry to describe a background data set

is to make sure the target in question does not become a background endmember. In the event that happens, then one will never find the target of interest because it is now part of the background description. This area was studied extensively in this research. The conclusion was that the masking idea presented did not necessarily prevent target pixels from becoming endmembers. Perhaps a new approach is needed that takes advantage of information we have about the target, which is in the form of target spaces. The cause of this may also be related to noise and calibration issues discussed previously. This area definitely needs further investigation.

Another suggestion would be to make the entire process presented in this research more "operational". That is, perhaps the process could be used in a near real time environment. If look-up-tables were constructed ahead of time, then collected radiance data could be entered directly into the algorithm for near real time processing. If the target spaces or look-up-tables are designed for the un-calibrated sensor data, then the need for sensor calibration is eliminated, thus leading to an increase in processing time. The logistics of such a process would need to be further investigated.

5.3.2 Detection

This research has developed a detection scheme that utilized physics based modeling as its input. We illustrated that the detection process worked and showed promise. However, the analysis of the final algorithm (PB-SIFT) was based on a more qualitative approach than a quantitative one. Based on the SIP behavior testing studies, a 2 dimensional decision criteria needs to be established. This could be in the form of a linear or non-linear boundary within the space. This decision boundary needs to be investigated as to its appropriateness and performance.

Another idea that may potentially work in the decision space is to not draw decision boundaries, but rather cluster the data. Since the data tend to fall in groups perhaps we could treat this space as a simple 2 dimensional classification problem. This is an entire field of study and could be investigation as to its appropriateness and potential performance.

During this research we developed the PB-SIFT algorithm to resemble that of ENVI's MTMF. We also presented a derivation of the latter algorithm, the MTMF. Future efforts should validate this derivation by coding it and testing it against the original implementation of ENVI's MTMF.

The concept and performance of the PB-SIFT algorithm should be tested against other detectors, both geometric and stochastic, some of which have been presented in the background section of this research. Additionally, larger data sets should be used so as to obtain possible false alarm rates on the order of 10^{-6} to 10^{-7}. This would require images to have 10^6 to 10^7 pixels in them.

Lastly, the concept of the SIP algorithm could probably be used as an added infeasibility metric to any geometric type detector. This could be look at and evaluated, since the metric doesn't necessarily require a target space, or matrix of vectors, as its input.

Singular Value Decomposition

The Singular Value Decomposition (SVD) is a widely used technique to decompose a matrix into several component matrices, exposing many of the useful and interesting properties of the original matrix. The decomposition of a matrix is often called a *factorization*. Ideally, the matrix is decomposed into a set of factors (often orthogonal or independent) that are optimal based on some criterion. For example, a criterion might be the reconstruction of the decomposed matrix. The decomposition of a matrix is also useful when the matrix is not of *full rank*. That is, the rows or columns of the matrix are linearly *dependent*. Theoretically, one can use Gaussian elimination to reduce the matrix to row echelon form and then count the number of nonzero rows to determine the rank. However, this approach is not practical when working in finite precision arithmetic. A similar case presents itself when using LU

decomposition where L is in lower triangular form with 1's on the diagonal and U is in upper triangular form. Ideally, a *rank-deficient* matrix may be decomposed into a smaller number of factors than the original matrix and still preserve all of the information in the matrix. The SVD, in general, represents an expansion of the original data in a coordinate system where the covariance matrix is diagonal.

Using the SVD, one can determine the dimension of the matrix range or more-often called the *rank*. The rank of a matrix is equal to the number of linear *independent* rows or columns. This is often referred to as a *minimum spanning set* or simply a *basis*. The SVD can also quantify the sensitivity of a linear system to numerical error or obtain a matrix inverse. Additionally, it provides solutions to least-squares problems and handles situations when matrices are either singular or numerically very close to singular.

The singular value decomposition provides a method for decomposing a rectangular, rather than a square, matrix. We can decompose a matrix that is not square nor symmetric by first considering a matrix $\mathbf{A}$ that is of dimension $m \times n$ where $m \geq n$. This assumption is made for convenience only; all the results will also hold if $m < n$. As it turns out, the vectors in the the expansion of $\mathbf{A}$ are the eigenvectors of the square matrices $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$. The former is a outer product and results in a matrix that is spanned by the row space of $\mathbf{A}$. The latter is a inner product and results in a matrix that is spanned by the column space (*i.e.*, the range) of $\mathbf{A}$.

The *singular values*, σ are the nonzero square roots of the eigenvalues (λ) from $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$ (*i.e.*, $\sigma = \sqrt{\lambda}$). The eigenvectors of $\mathbf{A}\mathbf{A}^T$ are called the "left" singular vectors ($\mathbf{U}$) while the eigenvectors of $\mathbf{A}^T\mathbf{A}$ are the "right" singular vectors ($\mathbf{V}$). By retaining the nonzero eigenvalues $k = \min(m, n)$, a *singular value decomposition* (SVD) can be constructed. That is

$$\mathbf{A} = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^T \tag{A.1}$$

where $\mathbf{U}$ is an $m \times m$ orthogonal matrix ($\mathbf{U}^T\mathbf{U} = \mathbf{I}$), $\mathbf{V}$ is an $n \times n$ orthogonal matrix ($\mathbf{V}^T\mathbf{V} = \mathbf{I}$), and $\mathbf{\Lambda}$ is an $m \times n$ matrix whose off-diagonal entries are all 0's and whose diagonal elements satisfy

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n \geq 0. \tag{A.2}$$

It can be shown that the rank of $\mathbf{A}$ equals the number of nonzero singular values and that the magnitudes of the nonzero singular values provide a measure of how close $\mathbf{A}$ is to a matrix of lower rank. That is, if $\mathbf{A}$ is nearly rank deficient (singular), then the singular values will be small. In general, the SVD represents an expansion of the original data $\mathbf{A}$ in a coordinate system where the covariance matrix $\mathbf{\Sigma}_A$ is diagonal.

Remember, this is called the singular value decomposition because the factorization finds values or eigenvalues or *characteristic roots* (all the same) that make the the following *characteristic equation* true or singular. That is

$$|\mathbf{A} - \lambda\mathbf{I}| = 0. \tag{A.3}$$

Using the determinant this way helps solve the linear system of equations thus generating an nth degree polynomial in the variable λ. This polynomial, that yields n-roots, is called the *characteristic polynomial.*

Equation (A.3) actually comes from the more *generalized eigenvalue equation* which has the form

$$\mathbf{A}\mathbf{x} = \lambda\mathbf{x} \tag{A.4}$$

which, when written in matrix form, is expressed as

$$\mathbf{AX} = \mathbf{X\Lambda}. \tag{A.5}$$

This implies

$$\mathbf{Ax} - \lambda\mathbf{x} = 0 \tag{A.6}$$

or

$$(\mathbf{A} - \lambda\mathbf{I})\mathbf{x} = 0. \tag{A.7}$$

The theory of simultaneous equations tells us that for this equation to be true it is necessary to have either $\mathbf{x} = 0$ or $|\mathbf{A} - \lambda\mathbf{I}| = 0$.

B

Maximum Noise Fraction (MNF)

B.1 Maximum Noise Fraction (MNF)

As a starting point principal component analysis (PCA) does not always produce components of decreasing image quality. This is because the PC approach relies on maximizing the band-to-band *total image variance*. It does not have the ability to separate signal and noise components, for example. A better metric of evaluation would be to maximize the signal relative to the noise (*i.e.*, signal-to-noise ratio (SNR)). This would, in turn, order the data in terms of *image quality* not image variance. The approach that performs this is called the maximum noise fraction (MNF) [12]. This algorithm establishes a noise fraction based on the ratio of the noise variance to the total variance for a particular band. It then generates a

linear transform such that the fraction is maximized. An important characteristic of the MNF transformation, which is not shared by the PC transformation, is the invariability to linear scaling. This is because we are maximizing a ratio (*i.e.*, signal-to-noise) not a (potentially scaled) variance.

B.2 Noise Adjusted Principal Components (NAPC)

An alternate and equivalent approach to the MNF transform is the noise adjusted principal component transform (NAPC) [30]. The MNF transform can be thought of as a transformation of the original data set to a space in which the covariance of the noise data is diagonal or uncorrelated, followed by a standard principal component transform. That is, the noise is arranged such that it is the same in every band. This is called *noise whitening*. A principal component transform on this data would then order the transformed data results in terms of relevant signal information.

In developing the theory, we assume that the signal and noise components in the data are uncorrelated. In doing so, we can write

$$\mathbf{\Sigma} = \mathbf{\Sigma}_S + \mathbf{\Sigma}_N \tag{B.1}$$

$$\mathbf{\Sigma}_S = \mathbf{\Sigma} - \mathbf{\Sigma}_N \tag{B.2}$$

where $\mathbf{\Sigma}$ is the covariance of the image data set, $\mathbf{\Sigma}_S$ is the signal covariance, and $\mathbf{\Sigma}_N$ is the noise covariance. We can then write the signal-to-noise ratio as

$$\begin{aligned} SNR &= \frac{\mathbf{e}_i^T \mathbf{\Sigma}_S \mathbf{e}_i}{\mathbf{e}_i^T \mathbf{\Sigma}_N \mathbf{e}_i} = \frac{\mathbf{e}_i^T \mathbf{\Sigma}\, \mathbf{e}_i}{\mathbf{e}_i^T \mathbf{\Sigma}_N \mathbf{e}_i} - \frac{\mathbf{e}_i^T \mathbf{\Sigma}_N \mathbf{e}_i}{\mathbf{e}_i^T \mathbf{\Sigma}_N \mathbf{e}_i} \\ &= \frac{\mathbf{e}_i^T \mathbf{\Sigma}\, \mathbf{e}_i}{\mathbf{e}_i^T \mathbf{\Sigma}_N \mathbf{e}_i} - 1 \end{aligned} \tag{B.3}$$

where $\mathbf{e}_i^T$ is a vector that maximizes the ratio. We also need the bands to be uncorrelated so that

$$\mathbf{e}_i^T \boldsymbol{\Sigma}\, \mathbf{e}_j = 0, \qquad i \neq j \tag{B.4}$$

Let $\mathbf{E}$ be the eigenvector matrix (computed from the noise covariance matrix) that digitalizes or decorrelates, $\boldsymbol{\Sigma}_N$ such that

$$\mathbf{E}^T \boldsymbol{\Sigma}_N \mathbf{E} = \boldsymbol{\Lambda}_N \tag{B.5}$$

where $\boldsymbol{\Lambda}_N$ is a matrix containing eigenvalues on the diagonal. Additionally, the orthonormal vectors $\mathbf{E}$ have the property $\mathbf{E}^T\mathbf{E} = \mathbf{I}$. Thus far, we have de-correlated the noise component. We now wish to find an operator that will produce bands with the same noise in each band. This is called *whitening* the data. This can be done by normalizing $\mathbf{E}$ by the standard deviation (or the square-root of the eigenvalues or variances) matrix. That is

$$\mathbf{F} = \mathbf{E}\boldsymbol{\Lambda}_N^{-1/2} \tag{B.6}$$

where

$$\begin{aligned} \mathbf{F}^T\mathbf{F} &= (\mathbf{E}\boldsymbol{\Lambda}_N^{-1/2})^T\, \mathbf{E}\boldsymbol{\Lambda}_N^{-1/2} = (\boldsymbol{\Lambda}_N^{-1/2})^T\, \mathbf{E}^T\mathbf{E}\boldsymbol{\Lambda}_N^{-1/2} \\ &= (\boldsymbol{\Lambda}_N^{-1/2})^T\boldsymbol{\Lambda}_N^{-1/2} = \boldsymbol{\Lambda}_N^{-1/2}\boldsymbol{\Lambda}_N^{-1/2} \\ &= \boldsymbol{\Lambda}_N^{-1} \end{aligned} \tag{B.7}$$

and

$$\begin{aligned} \mathbf{F}^T\boldsymbol{\Sigma}_N\mathbf{F} &= (\mathbf{E}\boldsymbol{\Lambda}_N^{-1/2})^T\boldsymbol{\Sigma}_N\, \mathbf{E}\boldsymbol{\Lambda}_N^{-1/2} = (\boldsymbol{\Lambda}_N^{-1/2})^T\mathbf{E}^T\boldsymbol{\Sigma}_N\mathbf{E}\boldsymbol{\Lambda}_N^{-1/2} \\ &= (\boldsymbol{\Lambda}_N^{-1/2})^T\boldsymbol{\Lambda}_N\boldsymbol{\Lambda}_N^{-1/2} = \boldsymbol{\Lambda}_N^{-1/2}\boldsymbol{\Lambda}_N\boldsymbol{\Lambda}_N^{-1/2} \\ &= \mathbf{I} \end{aligned} \tag{B.8}$$

We now introduce a set of vectors $\mathbf{z}$ such that

$$\mathbf{z}_i = \mathbf{F}^{-1}\mathbf{e}_i \tag{B.9}$$

which is re-expressed as

$$\mathbf{F}\mathbf{z}_i = \mathbf{e}_i \tag{B.10}$$

This can be substituted back into our SNR relation of Eq. (B.3) which yields

$$\begin{aligned} SNR+1 &= \frac{\mathbf{e}_i^T\boldsymbol{\Sigma}\,\mathbf{e}_i}{\mathbf{e}_i^T\boldsymbol{\Sigma}_N\,\mathbf{e}_i} = \frac{(\mathbf{F}\mathbf{z}_i)^T\boldsymbol{\Sigma}\,\mathbf{F}\mathbf{z}_i}{(\mathbf{F}\mathbf{z}_i)^T\boldsymbol{\Sigma}_N\,\mathbf{F}\mathbf{z}_i} = \frac{\mathbf{z}_i^T\mathbf{F}^T\boldsymbol{\Sigma}\,\mathbf{F}\mathbf{z}_i}{\mathbf{z}_i^T\mathbf{F}^T\boldsymbol{\Sigma}_N\,\mathbf{F}\mathbf{z}_i} \\ &= \frac{\mathbf{z}_i^T(\mathbf{F}^T\boldsymbol{\Sigma}\,\mathbf{F})\mathbf{z}_i}{\mathbf{z}_i^T\mathbf{z}_i}. \end{aligned} \tag{B.11}$$

We still wish to select $\mathbf{z}_i$ vectors that maximize (SNR +1) subject to the constraint $\mathbf{z}_i^T\mathbf{z}_i = 0$. The principal component of $\mathbf{F}^T\boldsymbol{\Sigma}\,\mathbf{F}$ will do this. That is, the $\mathbf{z}_i$ are the principal components of $\mathbf{F}^T\boldsymbol{\Sigma}\,\mathbf{F}$ and the eigenvalues of the transformed matrix are equal to (SNR +1).

B.2.1 Implementation

The implementation on the data set is then a three step process, a standard principal component (PC) transform, a noise whitening, and then another standard PC transform on the noise whitened data. Assuming matrix $\mathbf{A}$ is the image data, we have

$$\mathbf{P}_1 = \mathbf{E}^T\mathbf{A} \tag{B.12}$$

where $\mathbf{E}$ is the eigenvector matrix computed from the noise covariance, $\boldsymbol{\Sigma}_N$. Since it is assumed that the covariance of $\mathbf{A}$ has signal and noise components that are

uncorrelated, we have

$$\begin{aligned} \mathbf{\Sigma}_{\mathbf{P}_1} &= \mathbf{E}^T\mathbf{\Sigma}_S\mathbf{E} + \mathbf{E}^T\mathbf{\Sigma}_N\mathbf{E} \\ &= \mathbf{\Sigma}_{\mathbf{P}_1 S} + \mathbf{\Lambda}_{\mathbf{P}_1 N}. \end{aligned} \tag{B.13}$$

We then whiten $\mathbf{P}_1$ with $\mathbf{F} = \mathbf{E}\mathbf{\Lambda}_N^{-1/2}$ which is express as

$$\mathbf{P}_2 = \mathbf{F}^T\mathbf{P}_1 = (\mathbf{E}\mathbf{\Lambda}_N^{-1/2})^T\mathbf{P}_1. \tag{B.14}$$

This produces a covariance in the whitened space given by

$$\begin{aligned} \mathbf{\Sigma}_{\mathbf{P}_2} &= \mathbf{F}^T\mathbf{\Sigma}_{\mathbf{P}_1 S}\mathbf{F} + \mathbf{F}^T\mathbf{\Sigma}_{\mathbf{P}_1 N}\mathbf{F} \\ &= \mathbf{\Sigma}_{\mathbf{P}_2 S} + \mathbf{I}_N. \end{aligned} \tag{B.15}$$

where we take advantage of the result from Eq. (B.8).

Finally, we apply a standard principal components transform to the noise whitened data with eigenvector matrix $\mathbf{W}$ which is computed from the covariance $\mathbf{\Sigma}_{\mathbf{P}_2}$. This transformation, when applied to the data at this point, is express as

$$\mathbf{P}_3 = \mathbf{W}^T\mathbf{P}_2. \tag{B.16}$$

The covariance of $\mathbf{P}_3$ is then

$$\begin{aligned} \mathbf{\Sigma}_{\mathbf{P}_3} &= \mathbf{W}^T\mathbf{\Sigma}_{\mathbf{P}_2 S}\mathbf{W} + \mathbf{W}^T\mathbf{I}_N\mathbf{W} \\ &= \mathbf{\Lambda}_{\mathbf{P}_3 S} + \mathbf{I}_N \end{aligned} \tag{B.17}$$

since the orthonormal vectors $\mathbf{W}$ have the property $\mathbf{W}^T\mathbf{W} = \mathbf{I}$. This last transformation produces a diagonal covariance matrix where the noise is isotropic with unit variance.

Sum of Variances in a Linear Mixture

If we assume we have a mixing model of the form

$$s = \alpha t + (1 - \alpha)b \tag{C.1}$$

where α is a weighting scalar and t and b are random variables, then the variance of s is defined as

$$\sigma^2(s) = E[s^2] - E[s]^2 \tag{C.2}$$

where $E[\,]$ is the expected value. This expression can be factored as

$$\sigma^2(s) = E[(\alpha t + (1 - \alpha)b)^2] - E[\alpha t + (1 - \alpha)b]^2. \tag{C.3}$$

To simplify the notation, we let $\beta = (1 - \alpha)$. Then we have

$$\sigma^2(s) = E[(\alpha t + \beta b)^2] - E[\alpha t + \beta b]^2 \tag{C.4}$$

which reduces to

$$\begin{aligned}
\sigma^2(s) &= E[(\alpha t + \beta b)^2] - E[\alpha t + \beta b]^2 \\
&= E[\alpha^2 t^2 + 2\alpha\beta tb + \beta^2 b^2] - \Big(E[\alpha t] + E[\beta b]\Big)^2 \\
&= E[\alpha^2 t^2] + 2E[\alpha\beta tb] + E[\beta^2 b^2] - \Big\{(E[\alpha t] + E[\beta b])(E[\alpha t] + E[\beta b])\Big\} \\
&= E[\alpha^2 t^2] + 2E[\alpha\beta tb] + E[\beta^2 b^2] - \Big\{E[\alpha t]^2 + 2E[\alpha t]E[\beta b] + E[\beta b]^2\Big\} \\
&= E[\alpha^2 t^2] + 2E[\alpha\beta tb] + E[\beta^2 b^2] - E[\alpha t]^2 - 2E[\alpha t]E[\beta b] - E[\beta b]^2 \\
&= E[\alpha^2 t^2] - E[\alpha t]^2 + E[\beta^2 b^2] - E[\beta b]^2 + 2E[\alpha\beta tb] - 2E[\alpha t]E[\beta b] \\
&= \alpha^2(E[t^2] - E[t]^2) + \beta^2(E[b^2] - E[b]^2) + 2\alpha\beta\big(E[tb] - E[t]E[b]\big) \\
&= \alpha^2\sigma^2(t) + \beta^2\sigma^2(b) + 2\alpha\beta\sigma^2(t,b) \qquad \text{(C.5)}
\end{aligned}$$

using the expectation definitions of variance and covariance. Upon substitution of $\beta = (1 - \alpha)$ we have

$$\sigma^2(s) = \alpha^2\sigma^2(t) + (1-\alpha)^2\sigma^2(b) + 2\alpha(1-\alpha)\sigma^2(t,b). \qquad \text{(C.6)}$$

In the event we have independent random variables, then the covariance between t and b is zero. Therefore we have

$$\sigma^2(s) = a^2\sigma^2(t) + (1-a)^2\sigma^2(b). \qquad \text{(C.7)}$$

□

D

Geometry and the Linear Mixing Model (LMM)

The general equation for assuming a linear mixture of endmembers in a pixel is

$$\mathbf{x} = \mathbf{M}\mathbf{a} + \mathbf{w} \tag{D.1}$$

where $\mathbf{x}$ is the spectrum of the mixed pixel, $\mathbf{M}$ is a matrix of endmembers, $\mathbf{a}$ is a vector of endmember weights, and $\mathbf{w}$ is a vector of residuals accounting for noise and lack of fit. We wish to "unmix" the pixel of interest based on some number of endmembers. In order to do this we need to solve for the unknown weights, $\mathbf{a}$. The solution can thus be viewed as a linear estimation problem. If $\hat{\mathbf{a}}$ is an estimate of $\mathbf{a}$, then

$$\hat{\mathbf{x}} = \mathbf{M}\hat{\mathbf{a}} + \mathbf{w} \tag{D.2}$$

is an estimate of $\mathbf{x}$. The estimate of $\mathbf{a}$ can be found (in a least squares sense) by

$$\mathbf{M}^T\mathbf{x} = \mathbf{M}^T\mathbf{M}\mathbf{a} \tag{D.3}$$

$$\hat{\mathbf{a}} = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{x} \tag{D.4}$$

where the quantity $(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T$ is referred to as the pseudo-inverse of $\mathbf{M}$, which we will call $\mathbf{M}^\dagger$.

Rewriting Eq. (D.2) using the result from Eq. (D.4) we have

$$\hat{\mathbf{x}} = \mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{x} = \mathbf{M}\mathbf{M}^\dagger\mathbf{x}. \tag{D.5}$$

The quantity $\mathbf{M}\mathbf{M}^\dagger$ produces, what is called in matrix theory, a *projection matrix*, which we will denote by $\mathbf{P}$. This matrix projects any vector $\mathbf{x}$ onto the column space of $\mathbf{M}$. The projection matrix has two basic properties:

(i) It is idempotent: $\mathbf{P}^2 = \mathbf{P}$

(ii) It is symmetric: $\mathbf{P} = \mathbf{P}^T$

We can prove properties (i) and (ii) algebraically,

$$\begin{aligned}
\mathbf{P}^2 &= \mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T \\
&= \mathbf{M}\mathbf{I}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T \\
&= \mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T \\
&= \mathbf{P}
\end{aligned}$$

$$\begin{aligned}
\mathbf{P}^T &= \{\mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\}^T \\
&= (\mathbf{M}^T)^T[(\mathbf{M}^T\mathbf{M})^{-1}]^T\mathbf{M}^T \\
&= \mathbf{M}[(\mathbf{M}^T\mathbf{M})^T]^{-1}\mathbf{M}^T \\
&= \mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T \\
&= \mathbf{P}
\end{aligned}$$

Therefore, $\mathbf{P} = \mathbf{P}^T = \mathbf{P}^2$. We can rewrite Eq. (D.5) using our new projection matrix notation. That is,

$$\boxed{\hat{\mathbf{x}} = \mathbf{P}\mathbf{x}} \tag{D.6}$$

The error associated with the estimate of $\mathbf{x}$ is

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}. \tag{D.7}$$

Similarly, we can rewrite Eq. (D.7) in terms of our projection operator, $\mathbf{P}$.

$$\mathbf{e} = \mathbf{x} - \mathbf{P}\mathbf{x} = (\mathbf{I} - \mathbf{P})\mathbf{x} = (\mathbf{I} - \mathbf{M}\mathbf{M}^\dagger)\mathbf{x} \tag{D.8}$$

where we let $\mathbf{P}^\perp = (\mathbf{I} - \mathbf{M}\mathbf{M}^\dagger)$, which could be viewed as an orthogonal projection operator (which is also idempotent and symmetric). Finally, we have

$$\boxed{\mathbf{e} = \mathbf{P}^{\perp}\mathbf{x}} \tag{D.9}$$

Bibliography

[1] S.M. Adler-Golden, M.W. Matthew, L.S. Bernstein, R.Y. Levine, A. Berk, S.C. Richtsmeier, P.K. Acharya, G.P. Anderson, G. Felde, J. Gardner, M. Hike, L.S. Jeong, B. Pukall, J. Mello, A. Ratkowski, and H.H. Burke. Atmospheric correction for shortwave spectral imagery based on MODTRAN4. In *Proc. SPIE, Imaging Sspectrometry*, volume 3753, pages 61–69. SPIE, 1999.

[2] P. Bajorski, E.J. Ientilucci, and J.R. Schott. Comparison of basis-vector selection methods for target and background subspace as applied to subpixel target detection. In Sylvia S. Shen and Paul E. Lewis, editors, *Proc. SPIE, Algorithms and Technologies for Multispectral, Hyperspectra, and Ultraspectral Imagery X*, volume 5425, pages 97–108, Orlando, Fla, April 2004. SPIE.

[3] P. Bajorski, E.J. Ientilucci, and J.R. Schott. Geometric basis-vector selection methods and subpixel target detection as applied to hyperspectral imagery. In *Proc. of IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, Anchorage, Alaska, September 2004.

[4] Brent Bartlett. Sun-cloud spectrometer measurements, September 2005.

[5] J. W. Boardman. Leveraging the high dimensionality of AVIRIS data for improved sub-pixel target unmixing and rejection of false positives: mixture tuned matched filtering. In Robert O. Green, editor, *Summaries of the Seventh Annual JPL Air-*

borne Geoscience Workshop, volume 1 of *JPL Publication 97-21*, page 55, Pasadena, California, January 1998.

[6] J. W. Boardman, F. A. Kruse, and R. O. Green. Mapping target signatures via partial unmixing of AVIRIS data. In *Fifth JPL Airborne Earth Science Workshop*, volume 1 of *JPL Publication 95-1*, pages 23–26, Pasadena, California, 1995.

[7] Private conversation between John R. Schott and Joseph W. Boardman, 2004.

[8] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis.* Wiley, NY, 1973.

[9] W.H. Farrand and J.C. Harsanyi. Mapping the distribution of mine tailings in the Coeur d'Alene River Valley Idaho, through the use of a consstrained energy minimization technique. *Remote Sensing of Environment*, 59(1):64–76, 1997.

[10] Center for the Study of Earth from Space (CSES). *Atmosphere Removal Program (ATREM), Version 3.1, Users Guide.* University of Colorado, Boulder, 1999.

[11] R. Graham. An efficient algorithm for determining the convex hull of a finite point set. In *Info.Poc.Letters*, volume 1, pages 132–133, 1972.

[12] A. Green, M. Berman, P. Switzer, and M. Craig. A transformation for ordering multispectral data in terms of image quality with implications to noise removal. *IEEE Transactions on Geoscience and Remote Sensing*, 26(1):65–74, January 1988.

[13] J.C. Harsanyi and C.I. Chang. Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach. *IEEE Transactions on Geoscience and Remote Sensing*, 32(4):779–785, July 1994.

[14] G. Healey and D. Slater. Models and methods for automated material identification in hyperspectral imagery acquired under unknown illumination and atmospheric

conditions. *IEEE Transactions on Geoscience and Remote Sensing*, 37(6):2706–2717, November 1999.

[15] K.L. Hirsch, L. Balick, C. Borel, and McLachlan P. A comparison of four methods for determinging precipitable water vapor conten from multi-spectral data. In Sylvia S. Shen and Michael R. Descour, editors, *Proc. SPIE, Algorithms and Technologies for Multispectral, Hyperspectra, and Ultraspectral Imagery VII*, volume 4381, pages 417 – 428, Orlando, Fla, April 2001. SPIE.

[16] http://aviris.jpl.nasa.gov/.

[17] http://ltpwww.gsfc.nasa.gov/ISSSR 95/hydiceop.htm.

[18] http://www.cmdl.noaa.gov/infodata/ftpdata.html.

[19] http://www.cs.wisc.edu/condor/.

[20] R.A. Johnson and D.W. Wichern. *Apllied multivariate sstatistical analysis.* Prentice Hall, Upper Saddle River, NJ 07458, fourth edition, 1998.

[21] Y.J. Kaufman, D. Tanre, L.A. Remer, E.F. Vermote, A. Chu, and B.N Holben. Operational remote sensing of tropospheric aerosol over land from eos moderate imaging spectroradiometer. *J. Geopys. Res.*, 102(17051), 1997.

[22] Y.J. Kaufman, A.E. Wald, L.A. Remer, B.C. Gao, R.R. Li, and L. Flynn. The modis 2.1-m channel–correlation with visible reflectance for use in remote sensing of aerosol. *IEEE Transactions on Geoscience and Remote Sensing*, 35:1286–98, 1997.

[23] S.M. Kay. *Fundamentals of Statistical Signal Processing: Detection Theory.* Prentice-Hall signal processing series. Prentice-Hall PTR, Upper Saddle River, NJ 07458, 1998.

[24] E.J. Kelly. An adaptive detection algorithm. *IEEE Trans. Aerosp. Electon. Syst.*, 22(1):115–127, 1986.

[25] J.P. Kerekes and J.E. Baum. Spectral imaging system analytical model for subpixel object detection. *IEEE Transactions on Geoscience and Remote Sensing*, 40(5):1088–1101, 2002.

[26] M.D. King, W.P. Menzel, P.S. Grant, J.S. Myers, S.E. Arnold, G.T. Platnick, L.E. Gumley, S.C. Tsay, C.C. Moeller, M. Fitzgerald, K.S. Brown, and F.G. Osterwisch. Airborne scanning spectrometer for remote sensing of cloud, aerosol, water vapor and surface properties. *J. Atmos. Oceanic Technol.*, 13:777–794, 1996.

[27] S. Kraut and L.L. Scharf. The CFAR adaptive subspace detector is a scale-invariant GLRT. *IEEE Trans. Signal Process.*, 47(9):2538–2541, 1999.

[28] S. Kraut, L.L. Scharf, and L.T. McWhorter. Adaptive subspace detectors. *IEEE Trans. Signal Process.*, 49(1):1–16, 2001.

[29] F.A. Kruse, A.B. Lefkoff, J.W. Boardman, K.B. Heidebrecht, A.T. Shapiro, P.J. Barloon, and A.F.H. Goetz. The spectral image processing system (SIPS) - interactive visualization and analysis of imaging spectrometer data. *Remote Sensing of Environment*, 44:145–163, 1993.

[30] J. Lee, S. Woodyatt, and M. Berman. Enhancement of high spectral resolution remote sensing data by a noise-adjusted principal components transform. *IEEE Transactions on Geoscience and Remote Sensing*, 28(3):295–304, May 1990.

[31] K. Lee. *A sub-pixel scale target detection algorithm for hyperspectral imagery.* PhD dissertation, Rochester Institute of Technology, 54 Lomb Memorial Drive, Rochester, NY, 2003.

[32] D. Manolakis. Overview of algorithms for hyperspectral target detection: theory and practice. In *Proc. SPIE, Algorithms and Technologies for Multispectral,Hyperspectral, and Ultraspectral Imagery VIII*, volume 4725, pages 202–215, Orlando, FL, April 2002.

[33] D. Manolakis, D. Marden, and G.A. Shaw. Hyperspectral image processing for automatic target detection applications. *MIT Lincoln Laboratory Journal*, 14(1):79–116, 2003.

[34] D. Manolakis and G. Shaw. Detection algorithms for hyperspectral imaging applications. *IEEE Signal Processing Magazine*, 19(1):29–43, January 2002.

[35] D. Manolakis, C. Siracusa, D. Marden, and G. Shaw. Hyperspectral adaptive matched filter detectors: Practical perfomance comparison. In Sylvia S. Shen, editor, *Proc. SPIE, Algorithms and Technologies for Multispectral,Hyperspectral, and Ultraspectral Imagery VII*, volume 4381, pages 18–33, Orlando, FL, April 2001.

[36] D. Manolakis, C. Siracusa, and G. Shaw. Hyperspectral sub-pixel target detection using the linear mixing model. *IEEE Transactions on Geoscience and Remote Sensing*, 39(7):1392–1409, July 2001.

[37] P. Masson and W. Pieczynski. SEM algorithm on unsupervised statistical segmentation of stellite images. *IEEE Transactions on Geoscience and Remote Sensing*, 31(3):618–633, May 1993.

[38] R.A. McDonald. Opening the cold war sky to the public: declassifying satellite reconnaissance imagery. *Photogrammetric Engineering and Remote Sensing*, 61(4):385–390, 1995.

[39] F. Preparata and S.J. Hong. Convex hulls of finite sets of points in two and three dimensions. In *Comm. ACM*, volume 20, pages 87–93, 1977.

[40] (RSI) Research Systems Inc. *ENVI's User's Guide*, 2000.

[41] F.C. Robey, D.R. Fuhrmann, E.J. Kelly, and R. Nitzberg. A CFAR adaptive matched filter detector. *IEEE Trans. Aserosp. Electon. Syst.*, 38(1):208–216, 1992.

[42] L. Scharf and B. Friedlander. Matched subspace detectors. *IEEE Transactions on Signal Processing*, 42(8):2146–2157, August 1994.

[43] A. Schaum and A. Stocker. Spectral detection methods: Spectral unmixing, correlation processing, and when they are appropriate. In *Proceedings of the second annaul symposium on spectral sensing research*, July 1994.

[44] A. Schaum and A. Stocker. Spectral-selective target detection. In *Proceedings of ISSSR*, 1997.

[45] D. Schlapfer, C.C. Borel, J. Keller, and K.I. Itten. Atmospheric precorrected differential absorption technique to retrieve columnar water vapor. *Remote Sensing of Environment*, 65(3):353–366, 1998.

[46] J.R. Schott. *Remote Sensing: The Imaging Chain Approach.* Oxford University Press, New York, 1997.

[47] J.R. Schott. Hyperspectral algorithms course notes. Class Notes, 2002.

[48] J.R. Schott, K. Lee, R. Raqueno, G. Hoffmann, and G. Healey. A sub-pixel target detection technique based on the invariance approach. To be published, 2004.

[49] G. Shaw and D. Manolakis. Signal processing for hyperspectral image exploitation. *IEEE Signal Processing Magazine*, 19(1):12–16, January 2002.

[50] D. Slater and G Healey. Physics-based model acquisition and identification of airborne spectral images. In *Eighth IEEE International Conference on Computer Vision*, volume 2, pages 257–262. IEEE, July 2001.

[51] A. Stocker and A. Schaum. Application of stochastic mixing models (SMM) to hyperspectral detection problems. In Sylvia S. Shen, editor, *Algorithms for Multispectral and Hyperspectral Imagery III*, volume 3071, pages 47–60. SPIE, August 1997.